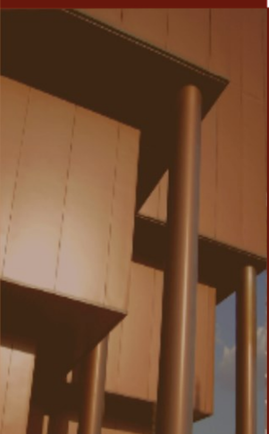
Trabajo Fin de Grado
Grado en Ingeniería Electrónica, Robótica y Mecatrónica

Power management strategies for a mobile robot

Autor: Francisco Javier Almécija Murciano

Tutor: Carlos Bordons Alba



Dpto. Ingeniería de Sistemas y Automática
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2019

Trabajo de Fin de Grado
Grado en Ingeniería Electrónica, Robótica y Mecatrónica

Power management strategies for a mobile robot

Autor:
Francisco Javier Almécija Murciano

Tutor:
Carlos Bordons Alba
Profesor de Universidad

Dpto. de Ingeniería de Sistemas y Automática
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla
Sevilla, 2019

Trabajo Fin de Grado: Power management strategies for a mobile robot

Autor: Francisco Javier Almécija Murciano

Tutor: Carlos Bordons Alba

El tribunal nombrado para juzgar el trabajo arriba indicado, compuesto por los siguientes profesores:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2019

El Secretario del Tribunal

To all the people I love

Acknowledgments

I would like to thank everybody that supported me during these years, I could not have done it without them.

In the first place I will thank my tutor, Carlos Bordons, because he was always there to solve my doubts and guide me over these months.

I must thank my friends because these years would not have been the same without them because they were the main reason why I enjoyed so much living in Seville and Bologna. Specially my roommates, who always encouraged each other when we were studying. Thank you, guys.

Thanks to my aunts, Montse and Amparo, that were always there when I needed them. Thanks to my grandmother who offered me her house to live in while I was studying. Thanks to my brothers, I really enjoyed the time we spent studying together and I hope you accomplish all your goals. To my grandfather, who have always been my role model.

I am especially grateful to my parents, whose efforts and sacrifices made possible that I study in other city and encouraged me to follow my dreams even in my lowest moments.

Finally, I would like to thank Anabel. Because she has been my greatest support in these last months when I thought I would never finish, and she was the one who encouraged and even force me to keep working hard until I finished this project.

Francisco Javier Almécija Murciano

Sevilla, 2019

Abstract

This document is part of the IUFCV Project, where the University of Sevilla, CSIRO (Commonwealth Scientific and Industrial Research Organization) and INTA (Instituto Nacional de Técnica Aeroespacial) are working together with unmanned mobile hybrid robots to improve its autonomy using $LiFePO_4$ batteries with PEM fuel cells.

The first goal of the project below, is modelling the 'Summit XL' UGV used in the IUFCV project, and more specifically its power supply, so three power management control strategies can be simulated in order to compare its performances.

This project is about designing the controllers to optimize the power management of the battery and the fuel cell.

Index

LIST OF TABLES

LIST OF FIGURES

1 INTRODUCTION AND OBJECTIVES

I believe that water will one day be employed as fuel, that hydrogen and oxygen which constitute it, used singly or together, will furnish an inexhaustible source of heat and light, of an intensity of which coal is not capable.

JULES VERNE

The project "Improving efficiency and operational range in low-power unmanned vehicles through the use of hybrid fuel-cell power systems (IUFCV)" aims to demonstrate and evaluate the technical feasibility of hybrid power systems, based on batteries and fuel cells, in existing unmanned vehicles.

It is a current project developed by the Laboratorio de Energía de CEDEA, in Instituto Nacional de Técnica Aeroespacial (INTA), from Huelva, Spain; the Autonomous Systems Lab, in Commonwealth Scientific and Industrial Research Organization (CSIRO), from Brisbane, Australia; and the Departamento de Ingeniería de Sistemas y Automática, in Universidad de Sevilla (DISA-US), from Seville, Spain. It is supported by The NATO Science for Peace and Security Programme.

The proposed hybrid power systems are designed and developed according to the specifications of three existing unmanned platforms, one autonomous underwater vehicle (AUV), also called unmanned underwater vehicle (UUV), and two unmanned ground vehicles (UGVs). These power systems will be integrated and evaluated in real operating conditions.

Unmanned ground vehicles are devices that operate by their own, without the influence of human beings, taking decisions (by obtaining information from the surroundings with sensors) of how to move and adapt to the environment around it. Unmanned underwater vehicles control techniques are normally more complex, as they have 3 more degrees of freedom of movement (6 against the 3 degrees of freedom that UGVs have).

In this project, the responsibility of the DISA-US is the design and implementation of the monitoring, control and energy management system onboard of the unmanned platforms.

This TFG in particular, is about the implementation of three different types of controllers to manage the hybrid power supply technology (with a PEM fuel cell and its $LiFePO_4$ battery) of the unmanned terrestrial robot Summit XL. This technology gives great supply stability, making it continuous, adds longer autonomy, high efficiency and practically zero environmental impact.

The use of PEM fuel cells and batteries hybrid technology means a big advantage over other type of energy, since fossil fuels are the most common source of energy for vehicles nowadays. Therefore, understanding the danger of fossil fuels is important to truly measure their impact on our lives and the life of our planet. That is why it is essential to make a change and start elaborating a new future of energy production based on renewable energies, zero-emissions sources, as well as energy saved through energy efficiency measures.

The energy management of robots is now an important field of research due to the usage of complementary sources of energy. The power supply management algorithms determine how the different elements of the hybrid vehicle power system (Fuel cell, batteries…) should operate in order to satisfy the power demand each time instant.

The main goal is the reduction of used energy, considering the constraints imposed by the vehicle and its components and satisfying the driver needs.

In hybrid vehicles it is necessary to determine which power sources to use each moment and how to manage the energy storage units.

In this project several simulations have been done in Matlab Simulink, where the models of the robot, the battery and the fuel cell have been implemented.

In this work three different approaches will be made to find the best control technique for FCHEV (Fuel Cell Hybrid Electric Vehicle):

- Heuristic control:
    - Heuristic strategy is based on intuitive rules and correlations involving various vehicular variables. For this project, the chosen guiding principle is that the battery discharge and charge phases should be regulated such that the SOC stays within predefined limits.
- ECMS control:
    - Equivalent consumption minimization strategy, also known as ECMS, is a strategy derived from Pontryagin's Minimum Principle and it is an online sub-optimal controller. Its main goal is the correct distribution of the power flows between the battery and the fuel cell, which are the sources supplying the demanded power.
- MPC control:
    - Model predictive control (MPC) is an advanced method of process control that is used to control a process while satisfying a set of constraints.

2 ENERGY SUPPLYING

In this project, the robot has a hybrid power supply technology (with a PEM fuel cell and its LiFePO4 battery) because a fuel cell hybrid electric vehicle (FCHEV) is more advantageous compared to a gasoline-powered ICE based vehicle or a traditional HEV because a FCHEV only uses one electric motor instead of an ICE or an electric motor combined with an ICE.

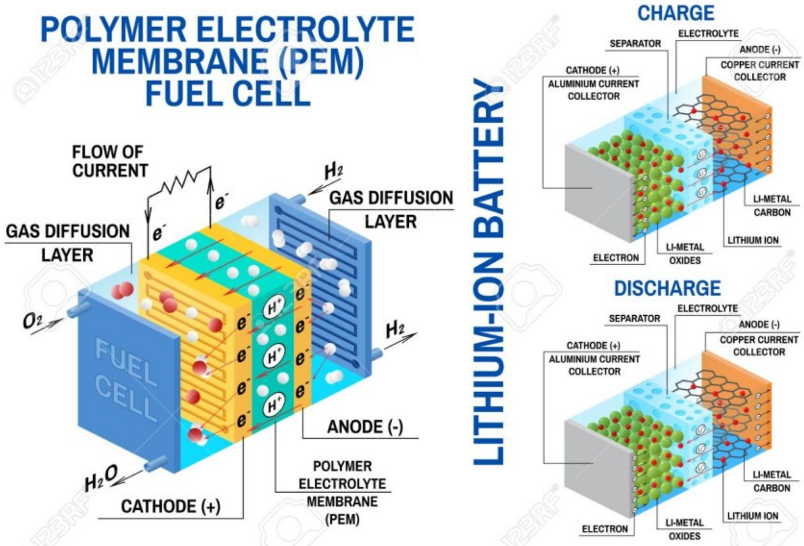


Figure 1 PEM Fuel cell and Li-ion battery diagrams

Batteries

A battery is a device that produces electrons through electrochemical reactions and contains positive and negative terminals. It is based on redox reactions: one of the components is oxidized (loses electrons) and the other one is reduced (gains electrons).

Batteries store chemical energy in one or more electrochemical cells and transform it into electrical energy. Each cell consists of an electrolyte (liquid or solid) together with a positive (anode) and negative (cathode) electrode. During discharge, the electrochemical reactions take place at the two electrodes and the electron current flows through the external circuit. This reaction is reversible, allowing the battery to recharge applying an extern voltage between both electrodes.

Some battery types include lithium-ion (Li-ion), nickel metal hydride (NiMH), nickel-zinc (NiZn), and nickel-cadmium (NiCd) cells. So far, Li-ion batteries have the highest market value and they are the most relevant ones for this project, as they are the ones used.

They consist of a Li metallic oxide cathode (in this case, LiFePO4), an electrolyte of lithium salts dissolved in organic carbonates, a carbon anode combined with lithium, and a separator.

2.1.1 Lithium-ion batteries

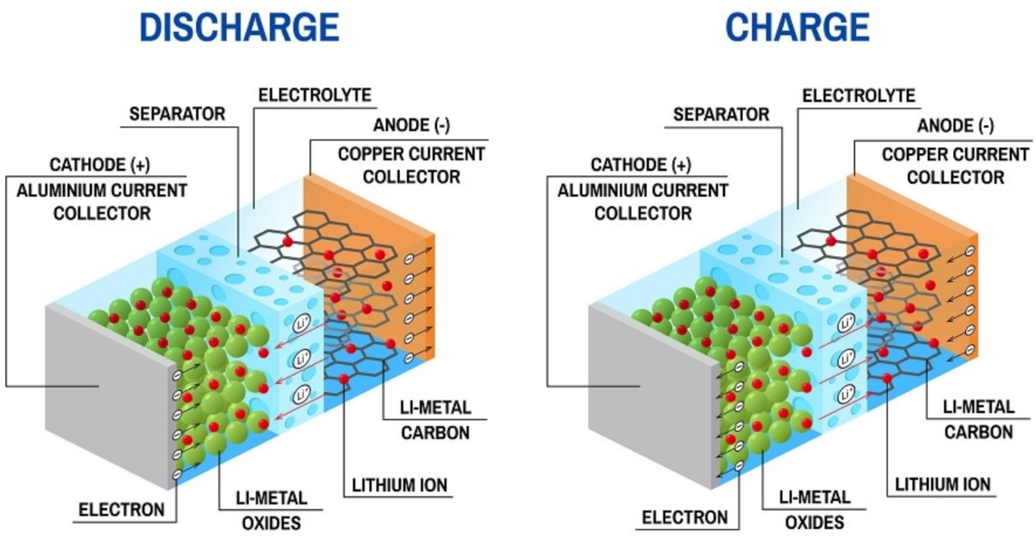


Figure 2 Lithium-ion battery diagram of charge and discharge

Lithium-ion batteries (LIB) are a family of rechargeable batteries having high energy density and commonly used in mobile or portable systems and in hybrid and electric vehicles.

Li-ion batteries are significantly lighter than other kinds of rechargeable batteries of similar size, that is why they are heavily used in portable electronics. These batteries can be commonly found in cell phones, laptops, etc.

When a LIB is discharging, lithium ions move from the negative electrode (anode) to the positive electrode (cathode). When a LIB is charging, lithium ions move in the opposite direction, and the negative electrode becomes the cathode, while the positive electrode becomes the anode.

Table 1 Advantages and disadvantages of Li-ion batteries [1].

ADVANTAGES	DISADVANTAGES
Higher energetical density (It can store 150 watt-hours electricity per kg).	LIBs start to degrade the moment they leave the factory. They usually last for only two to three years from the date of manufacture, regardless of whether used or unused.
Lower self-discharge rate. (They usually lose approximately 5% of their charge each month, against a 20% for NiMH).	LIBs are highly dependent to higher temperatures; this leads to a much faster degradation rate than normal.

LIBs do not require complete discharge prior to recharging.	If they are fully discharged, it gets totally damaged.
It can handle more charge/discharge cycles.	LIBs are comparatively expensive.
LIBs barely require maintenance.	There exists a small possibility that if the LIB pack fails, it may burst open into flame.

Fuel cells

A fuel cell (FC) is a device that converts chemical energy of a fuel into electricity. It consists of an electrolyte and two electrodes. Fuel cells generate the electrical power from a fuel (hydrogen) and an oxidant (oxygen). The fuel cell consumes oxygen from the air and hydrogen from a tank (or any other supplier). A chemical catalyst may be used to speed up the chemical reaction in the cell. It produces electricity without combustion and is hence less polluting.

The fuel cell was first devised by Sir William Grove in 1839. William Grove postulated that by reversing the electrolysis process, electricity and water could be produced. [2]

Fuel cells produce electricity by making use of chemical energy generated through a chemical reaction between positively charged ions and an oxidizing agent. They consist of an electrolyte and two electrodes. The positively charged electrode is called the anode and the negatively charged electrode is called the cathode. [2]

A fuel cell converts the chemical energy of the reaction between charged hydrogen and oxygen ions into electricity. The positively charged hydrogen cells move between the two electrodes to create a flow of electricity which is directed outside the cell to provide electricity. As long there is a flow of chemicals into the cell, it never goes dead, unlike conventional batteries which require recharging after a while. [2]

There are several types of fuel cells: alkaline, molten carbonate, methanol…However, in this work the FC used is Polymeter Electrolyte Membrane (PEM) due to their power density and low operating temperature.

2.2.1 PEM fuel cells

The polymer electrolyte membrane fuel cell (PEMFC), also known as proton exchange membrane fuel cell, takes its name from the type of electrolyte: a polymeric membrane with high proton conductivity when the membrane is conveniently hydrated. The operation of a PEM FC is briefly explained in **Figure 3**:

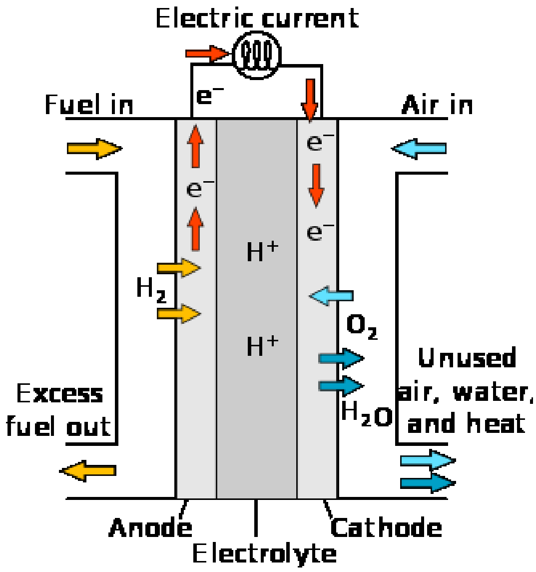


Figure 3 Simplified diagram of H_2 working PEMFC

In the anode, the hydrogen molecules are divided into protons (H+) and electrons. The protons pass the membrane to the anode, while the electron travel through the external circuit, producing current to the anode. In the anode, these protons and electrons react with the oxygen, producing water [3].

$$2H^{+} + 2e^{-} + \frac{1}{2}O_2 \rightarrow H_2O \qquad (1)$$

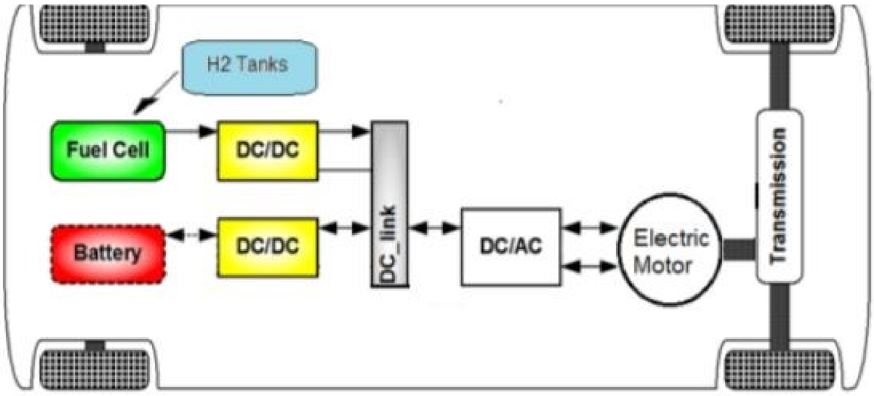
Table 2 Advantages and disadvantages of PEM FC [4].

ADVANTAGES	DISADVANTAGES
High efficiency compared with other energy conversion devices (Twice a gasoline vehicle's efficiency).	High cost of hydrogen.
Efficiency high with partial loads, unlike ICE.	High production cost of hydrogen.
Local emissions problem in densely urban areas can be eliminated.	
Low operation temperature (below 80ºC).	
Smaller cost of the materials (except for the catalyst, based on platinum).	
Operation is safer.	

Hybrid technology

A FCEV is a zero-emission vehicle with a fuel cell system that generates electricity to propel the vehicle and operate auxiliary equipment. Hydrogen fuel is consumed in the fuel cell stack to produce electricity, heat, and water vapor—no harmful pollutants are emitted from the vehicle. [5]

Like hybrid electric and battery electric vehicles, FCEVs also use traction batteries, inverters, and electric motors. FCEVs also require the use of electrified accessories, which are beginning to be developed to improve overall efficiency and reduce emissions. **Figure 4** FCEV basic configuration shows a basic representation of the major system components in a FCEV.



(Intechopen.com, undated)

Figure 4 FCEV basic configuration [5]

The energy storage system (batteries or capacitor) is connected via the converters to the electric motor that moves the vehicle. The fuel cell supplies energy to the electric motor and/or delivers power to the energy storage system.

The power conditioning requires power regulation and inversion. Fuel cells and batteries both produce direct current (DC) electricity, while the electric drivetrain may require alternating current (AC) or DC. A DC/DC converter regulates the fuel cell power. For AC electric drivetrains, the DC power must be inverted to power the electric motors, typically by using a DC/AC inverter.

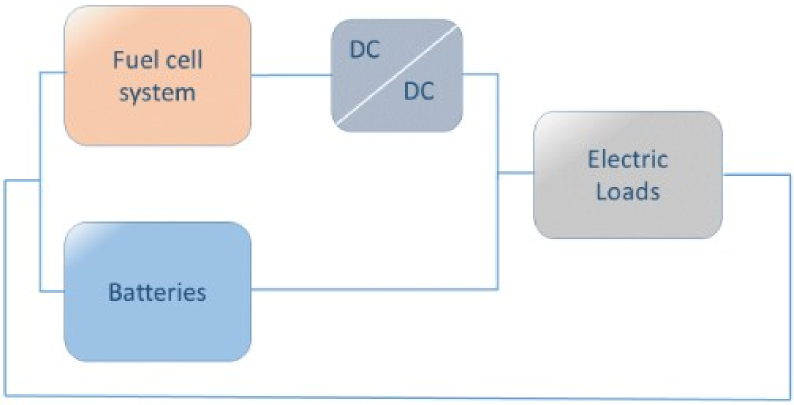


Figure 5 Active hybrid power system

Both batteries and fuel cells can power a charge (for example, an engine), but in real practise fuel cells are not used by their own. They are both combined in different configurations, as the ones shown in **Figure 5** and **Figure 6**.
The active configuration allows a decoupling of sizing and operating conditions in batteries and fuel cell, thanks to the DC/DC converters, allowing also a more precise control of the power system. The main disadvantages of indirect hybrids (active) are the more complex system topology, reduced efficiency due to losses at the voltage, system cost, and higher weight and volume [6].

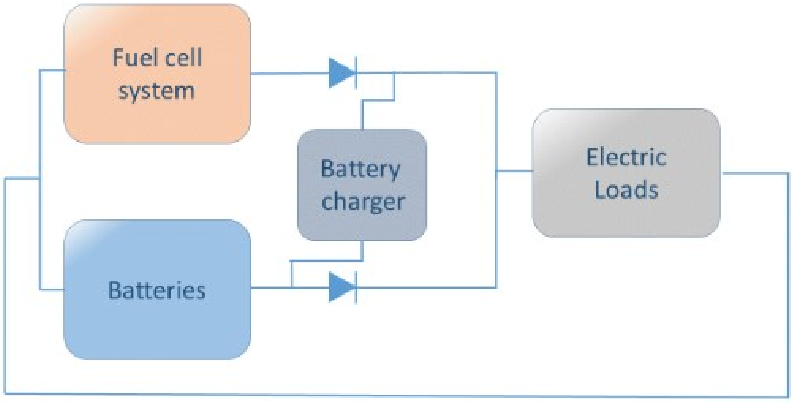


Figure 6 Passive hybrid power system

On the other hand, the passive configuration with direct connection to DC bus offer the advantages of lower losses, reduced cost and simple architecture. However, active power control is not possible, and a careful design and integration of fuel cells and batteries is required to ensure a similar voltage range operation and proper charging conditions of the batteries from fuel cell if this option is considered [5].

Table 3 Advantages and disadvanteges of hybrid configurations [6].

	Active configuration	**Passive configuration**
Advantages	Decoupling of sizing and operating conditions in batteries and fuel cell	Lower losses
	More precise control of the power system	Reduced cost
		Simpler architecture
Disadvantages	More complex system topology	Active power control is not possible
	Reduced efficiency due to losses at the voltage	Careful design and integration of fuel cells and batteries
	Higher system cost	
	Higher weight and volume	

In this project a passive hybrid configuration, with direct coupling between batteries and fuel cells has been chosen.

Examples of Fuel cell electric vehivles (FCEV)

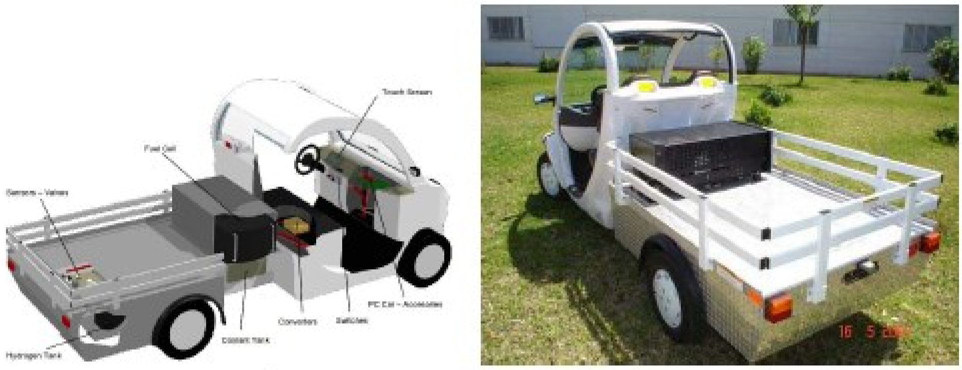
2.4.1 Delfin projects

The goal of these research projects is to prove the feasibility of the use of hydrogen as an energy source for automotive applications. For this aim, a commercial electric car (GEM eL, **Table 4**) was acquired as an experimental platform. This vehicle was used in two projects (Delfin I and II).

2.4.1.1 Delfin I

The original power train of the vehicle has been used. The power of the D.C. electric motor is 3.72 kW at 72

Volts, with 6 gel batteries of 12 V each. The PEM fuel cell is the HyPM-12XR, supplied by Hydrogenics, which main characteristics are shown in **Table 5**.



(a) Delfín I scheme. (b) Delfín I (vehicle).

Figure 7 Delfin I

Table 4 GEM eL specifications [3]

Curb Weight	1,285 lb
GVW	2,300 lb
Payload Capacity	1,015 lb
Length	144”
Height	70”
Width	55”
Wheelbase	114”
Cubic Feet of Cab	47 ft^3
Turning Radius	17 ft
Tires	12-inch
Top Speed	25 mph
Ground Clearance	8”

Table 5 Hydrogenics HyPM-12XR specifications [3]

Maximum output power	12.5 kW
Output voltage range	37-57 V
Maximum current	350 A
Dimension	90x50x32 cm
Volume	153.61
Weight	90kg

Hydrogen is stored on board at 200 bar using a tank supplied by Dynetek (model L033), 55 with a capacity of 5.8 Nm3, equivalent to 476.25 grams of hydrogen and 33 litres of volume.

The hydrogen storage system includes pressure sensors, electro valves, regulators and the outlet connection for refuelling. **Figure 7** shows a scheme of the disposition of the fuel tank and the rest of the devices of the vehicle and the real vehicle.

2.4.1.2 Delfin II

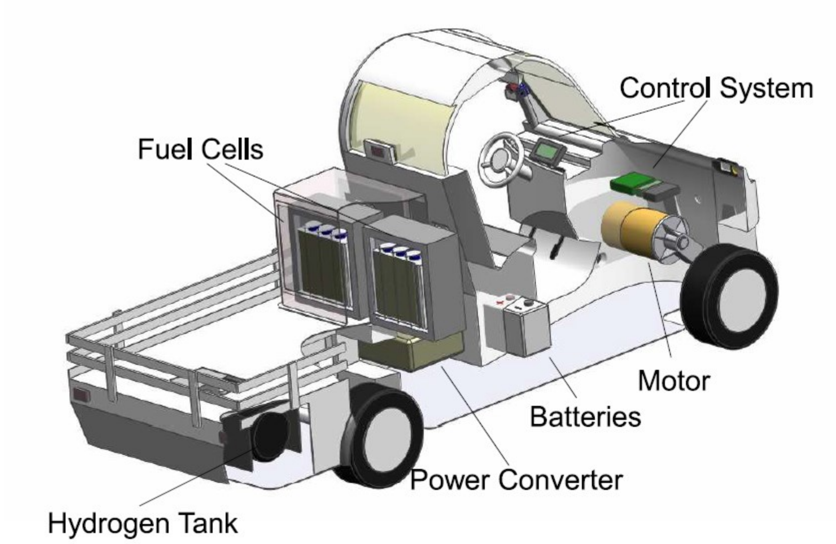


Figure 8 Delfin II

Based on the same vehicle as the Delf ́ın I project (a GEM eL electric car), the motor and the hydrogen storage system are the only elements remaining from the previous project.

Figure 8 shows the new disposition of the devices of the vehicle.

2.4.2 Hercules project

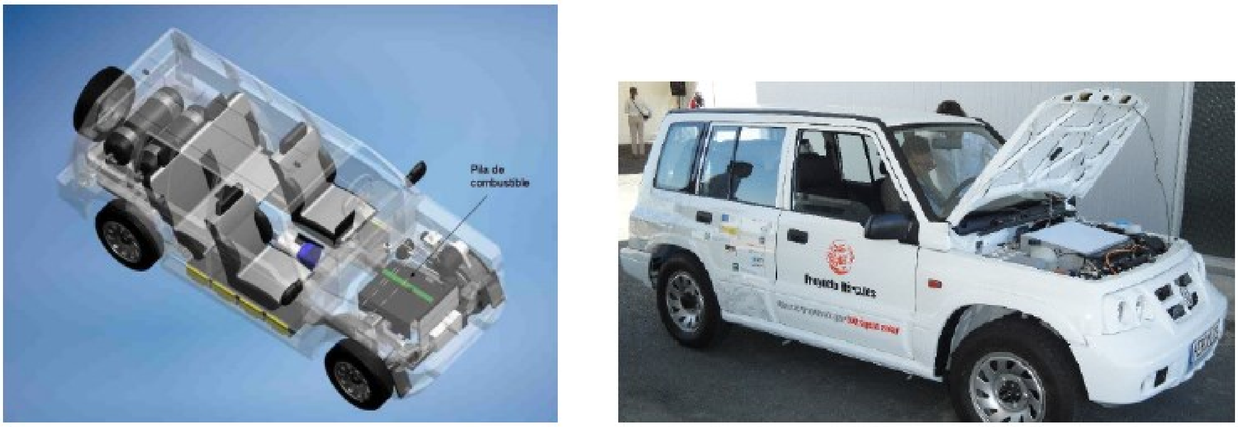


Figure 9 Hercules vehicle.

The vehicle is based on a commercial Santana 350 SUV. In this vehicle the engine and its auxiliary devices have been removed. In their place, the following devices have been installed:

- A PEM fuel cell: Nuvera with a maximum peak power of 56kW.
- A pack of Lithium-ion Batteries: four modules of 13 Li-ion 3.7 V cells in series, model Kokam SLPB 125255255H.
- A PermanentMagnet SynchronousMotor (PMSM): the nominal power is 66 kWand the maximum torque is 460Nm.
- A hydrogen storage system: it consists of two tanks of 33l and one of 24l, the three of them with a

maximum pressure of 350 bar. This system could store up to 2.4kg of hydrogen.

The possibility of substituting the batteries for UCs or combining both types of sources is also studied. In the first case, two modules in parallel of 126 Maxwell BCAP 2000 capacitors in series would substitute the pack of batteries. For the second, only two of the four modules of batteries and one of the two modules of UCs would be used [3] [7] [8].

A scheme of the Hercules vehicle is shown in **Figure 10**. The fuel cell and the lithium ion batteries feed an electrical motor through DC/DC converters to connect the different systems to the DC bus. The DC/DC converter which connects the fuel cell to the DC bus is unilateral and rises the fuel cell voltage to the DC bus voltage. The other two converters are bidirectional, allowing regenerative braking and battery recharging.

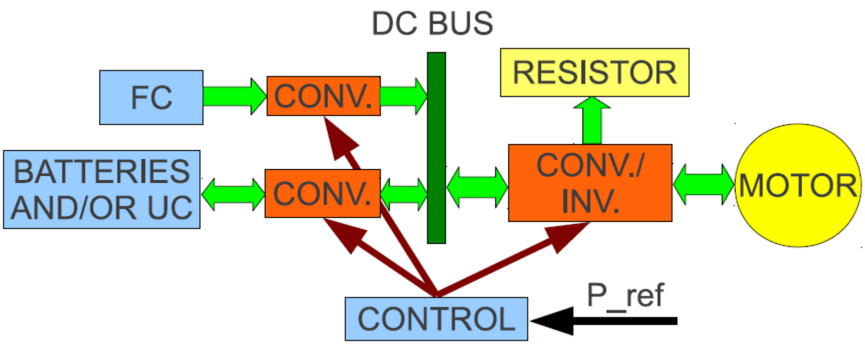


Figure 10 Hercules scheme.

2.4.3 IUFCV project

The project "Improving efficiency and operational range in low-power unmanned vehicles through the use of hybrid fuel-cell power systems (IUFCV)" aims to demonstrate and evaluate the technical feasibility of hybrid power systems, based on batteries and fuel cells, in existing unmanned vehicles.

It is a current project developed by the Laboratorio de Energía de CEDEA, in Instituto Nacional de Técnica Aeroespacial (INTA), from Huelva, Spain; the Autonomous Systems Lab, in Commonwealth Scientific and Industrial Research Organization (CSIRO), from Brisbane, Australia; and the Departamento de Ingeniería de Sistemas y Automática, in Universidad de Sevilla (DISA-US), from Seville, Spain. It is supported by The NATO Science for Peace and Security Programme.

The proposed hybrid power systems are designed and developed according to the specifications of three existing unmanned platforms, one autonomous underwater vehicle (AUV), also called unmanned underwater vehicle (UUV), and two unmanned ground vehicles (UGVs). These power systems will be integrated and evaluated in real operating conditions. The chosen criteria for success is the one shown in **Table 6**. [6]

Table 6 Criteria for success [6]

CRITERIA FOR SUCCESS
Specific energy of the fuel cell hybrid power systems > 180 Wh/kg (without O2 storage in the AUV)
Endurance of the fuel cell UGVs (in runtime-nominal usage) > 7 hours
Endurance of the fuel cell UUV > 10 hours
Recharging time < 5 minm (for hydrogen compressed gas)
Availability of the power system > 95%
Achievement of end user requirements
Application of existing RCS related to the safe use of hydrogen and fuel cells

This project involves three platforms: the underwater (UAV) Starbug which is placed in the CSIRO Laboratory, in Brisbane, Australia, as well as one of the ground devices (UGVs), called Husky; the last robot is called Summit XL and it is a commercial platform, from Robotnik, currently placed in the Universidad de Sevilla Department.

The objective of DISA-US with this last robot is to design the control for the energy management, making it efficient. This Project is about to design the controllers to optimize the power management of the battery and the fuel cell.

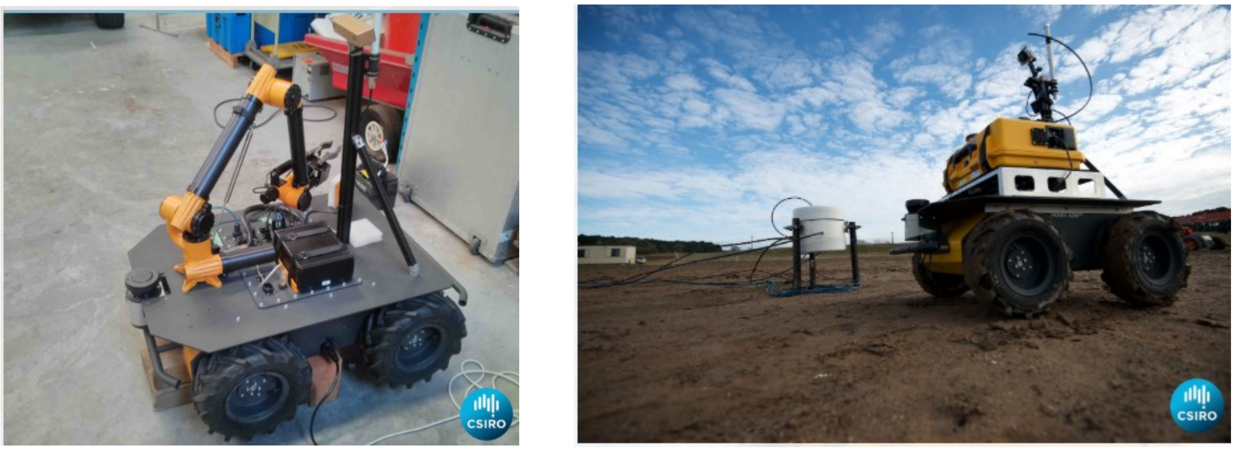


Figure 11 Husky [5]

Figure 13 Starbug X ROV [5]

Figure 12 Summit XL [5]

3 SUMMIT XL UNMANNED GROUND VEHICLE

In this chapter, the Summit XL UGV, from Robotnik will be introduced. It is the mobile robot whose power management is trying to be improved in this Project, because it is also the platform chosen by INTA and US to install the fuel cell for the IUFCV Project.

This platform is used for testing different power systems configurations and technologies, as well as to integrate sensors and simulate missions defined by other users, such as the Chemical, biological, radiological and nuclear (CBRN) and Materials Department of INTA [6].

The main characteristics of this platform are as follow [9] [10] [11] [12].

- Size: 722x613x392 mm
- Weight: 45 kg
- Max. payload: 20 kg
- Enclosure class: IP54
- Speed: 3 m/s
- Drive system: 4 wheel, 4x250W brushless motors
- Driver motor: DZCANTE 020L080 and MC1DZC board
- Camera: AXIS p5514 PTZ Dome Network Camera
- Sensor: Stick laser range Finder, amplitude 270º, 10m range
- Motherboard: Mitac PD10B1 MT con Quad core Intel Bay Trail J1900
- Autopilot: Pixhawk FPU PX4 (gyroscope and accelerometer)
- Controller: PS3 Bluetooth remote controller
- Batteries: 8x3.3V LiFePO4
- Router: Belkin N300 Wi-Fi N
- Autonomy: 5 hours (continuous usage), 20 hours (standard laboratory usage)

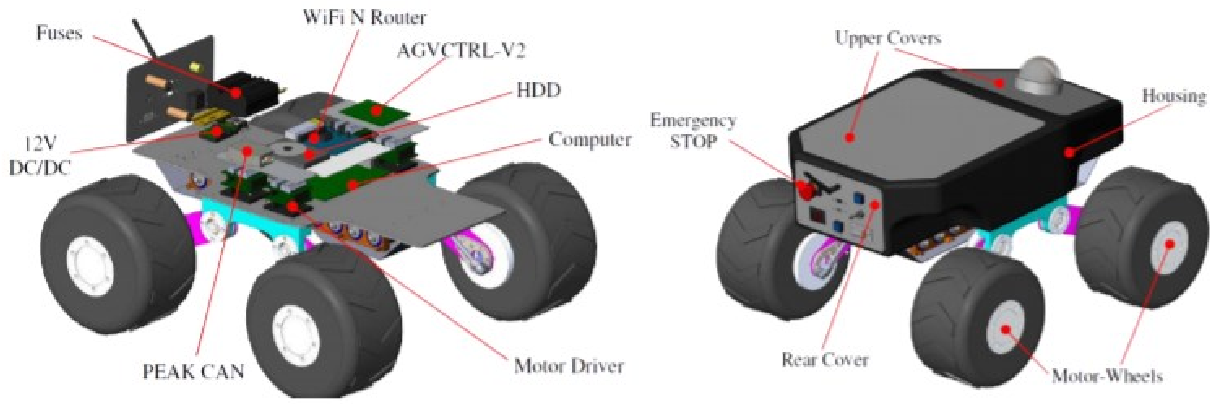


Figure 14 Summit XL scheme

(a) Front view.

(b) Back view.

(c) Lateral view.

Figure 15 Summit XL unmanned ground vehicle [13]

One of the innovations for the Summit XL in IUFCV Project would be to increase the autonomy up to seven hours of continuous usage, maintaining its core capabilities in terms of payload. It would have a power system based on open cathode and air cooled PEMFCs and Li-ion batteries. Besides, compressed hydrogen and metal hydrides will be the hydrogen storage technologies used in the platform. It would also have a passive hybrid configuration, with direct coupling between batteries and fuel cells [6].

The Summit XL has an embedded computer where runs the Ubuntu 14.04 LTS OS from Linux. In that computer ROS is installed in its version 2019 ROS Indigo. For compatibility reasons the same OS and version of ROS were installed in the DISA computer in order to monitor the transmitted data [9].

Energy Management System (EMS)

The EMS was designed to develop and implement a control and monitorization system for a hybrid power supply system consisting of a battery and a fuel cell in a passive configuration.

The scheme of the power bus of the Summit XL is shown in **Figure 16** [14]**.**

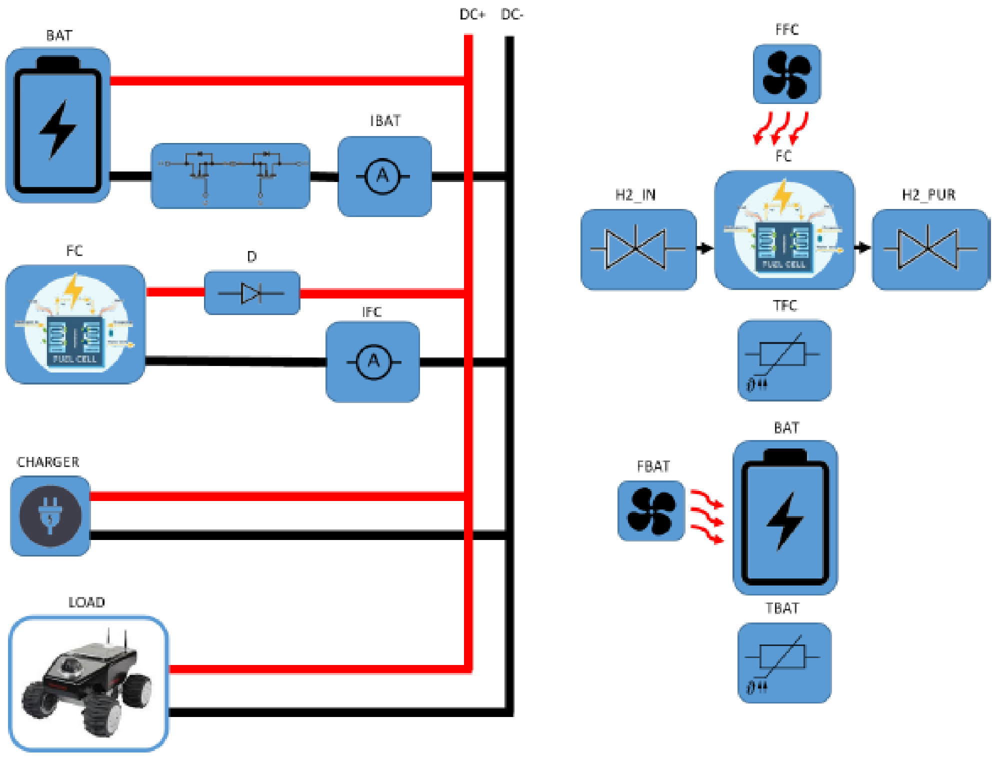


Figure 16 Hybrid system configuration [14].

The control system basic functionality is the protection and control of li-ion batteries (BMS), as well as the management and control of the fuel cell:

- Protection and control of Li-ion batteries: voltage reading, current reading, protection against overvoltage, overcurrent, low voltage and high temperature. To do this, the system implements a configuration based on very low series resistance Mosfet which allow to enable or disable the charge/discharge depending on the operative conditions.
- Control of the fuel cell: it includes every needed action for the management of the electro valves for supply and purge hydrogen, as well as the thermical management through the flow rate from the fan of the fuel cell. In addition, sensors for monitor the voltage and current of the fuel cell are included, as well as hydrogen tank pressure to estimate the autonomy.

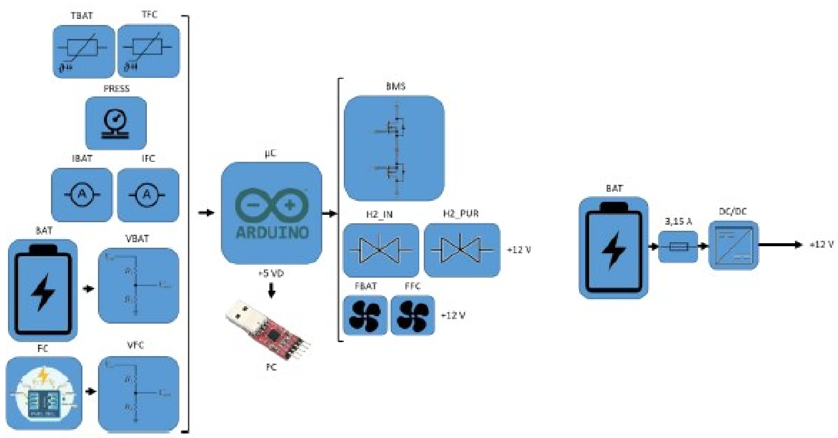


Figure 17 Conceptual scheme of the control board [14].

Table 7 Control parameters

	Output parameters
I_{BAT}	Battery current (A)
V_{BAT}	Battery voltage (V)
T_{BAT}	Battery temperature (ºC)
I_{FC}	Fuel cell current (A)
V_{FC}	Fuel cell voltage (V)
T_{FC}	Fuel cell temperature (ºC)
P_{FC}	Hydrogen tank pressure (Bar)
	Input parameters
C_{BAT}	Enable/Disable the charge of the battery
D_{BAT}	Enable/Disable the discharge of the battery
$H2_{IN}$	Enable/Disable the electro valve for hydrogen supply
PURGE	Enable/Disable the electro valve for hydrogen purge
FAN_{BAT}	Enable/Disable the fan of the battery
FAN_{FC}	Enable/Disable the fan of the fuel cell
FAN_{FC_SPEED}	Signal to control the speed of the fan of the fuel cell
	Intern parameters
OV	Failure due to overvoltage in the batteries
UV	Failure due to undervoltage in the batteries
OTC	Overtemperature failure during the charge of the battery
OTD	Overtemperature failure during the discharge of the battery
OC	Overcurrent of the battery
$ERROR_{H2}$	Failure due to lack of hydrogen supply
$ERROR_{T^{a}}$	Thermal failure in the fuel cell

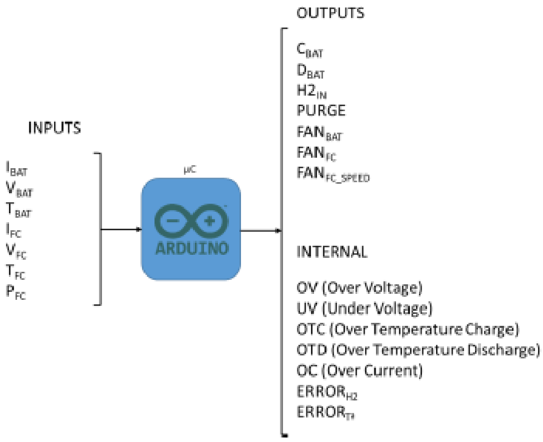


Figure 18 In/out Parameters [14].

Table 8 Operation mode of BMS

MODE	CHARGE	DISCHARGE	RECOVERY METHOD
Normal	ON	ON	-
UV	ON	OFF	Vcell>Vmin
OV, OT	OFF	ON	Vcell<Vmax/Tª<Tªmax
OC, OT	OFF	OFF	Auto/Tª<Tªmax

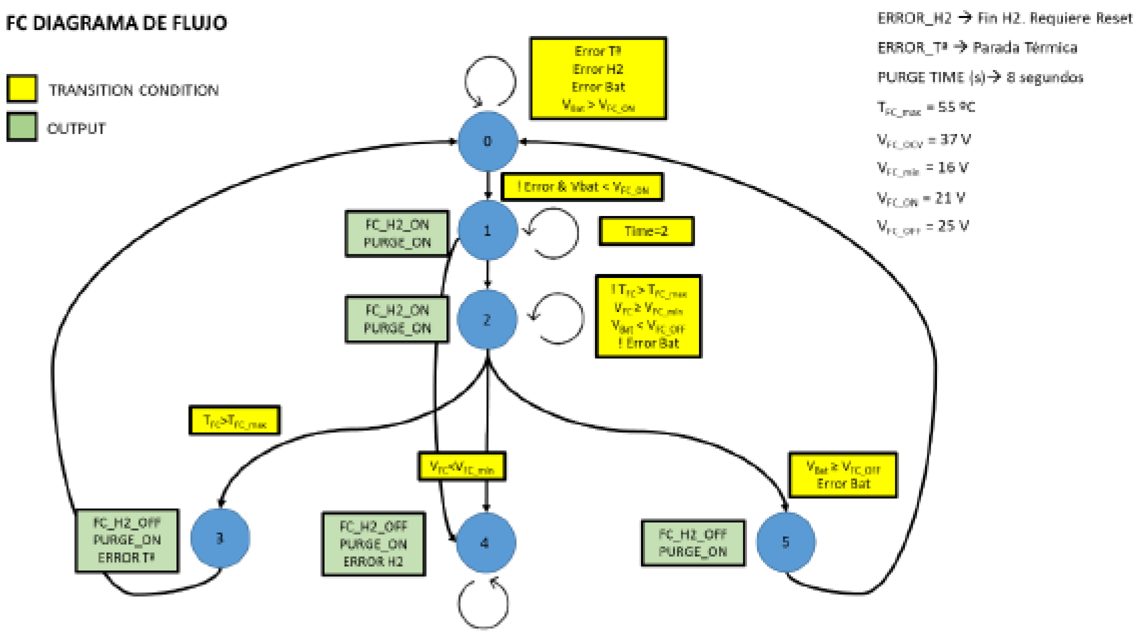


Figure 19 Diagram about fuel cell control [14].

4 MODELS

A model is a task-driven, purposeful simplification and abstraction of a perception of reality, shaped by physical, legal, and cognitive constraints. It is task-driven, because a model is captured with a certain question or task in mind. Simplifications leave all the known and observed entities and their relation out that are not important for the task. Abstraction aggregates information that is important, but not needed in the same detail as the object of interest. Both activities, simplification and abstraction, are done purposefully. However, they are done based on a perception of reality. This perception is already a model in itself, as it comes with a physical constraint. [7]

The construction of a model is a science that combines mathematics and logic. Generally, the experience shows that it is better to start with simple models that will be turning more and more detailed progressively.

The model should have the level of detail required to accomplish the goals of the study. Given a mathematical model, the construction of the computer model is indispensable to be able to manipulate numerically the model, so we can get the required information about the system.

In this Project, the chosen software to simulate the model of the UGV is Simulink, developed by MathWorks, is a graphical programming environment for modelling, simulating and analysing multidomain dynamical systems. Its primary interface is a graphical block diagramming tool and a customizable set of block libraries. It offers tight integration with the rest of the MATLAB environment and can either drive MATLAB or be scripted from it. Simulink is widely used in automatic control and digital signal processing for multidomain simulation and model-based design. [15]

Figure 20 Matlab & Simulink logo

For this study, an unmanned ground vehicle (Summit XL) with its power system based on a Li-ion battery and a PEM fuel cell had been modelled [3].

There are two ways of focus the modelling, quasistatic approximation and dynamic approximation:

- Quasistatic: The input variables are speed, acceleration and the inclination of the road where the vehicle is. With these parameters it is calculated the necessary force in the wheels to follow a driving cycle. It is assumed that for a little period of time, the vehicle has a constant speed, acceleration and inclination.
- Dynamic: It is based on a correct mathematical description of the system. Usually formulated using differential equations.

In this project a quasistatic approach has been considered.

Summit model

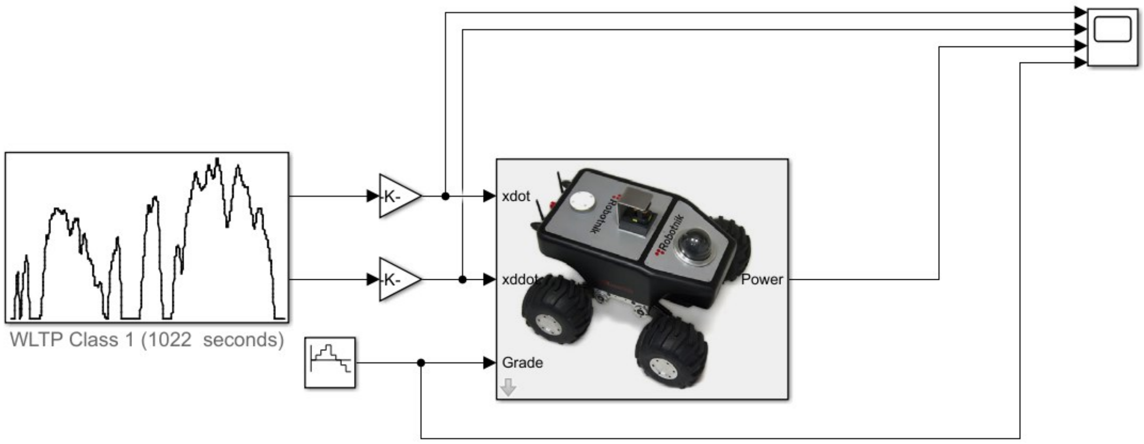


Figure 21 Quasistatic model of the Summit

As showed in **Figure 21**the model of the vehicle receives the speed, acceleration of a predefined driving cycle, in this case, WLTP has been chosen, but it has been scaled to require a maximum speed of 3 m/s, as it is the maximum speed of the robot. In addition, a Repeating sequence stair block has been included to simulate the inclination during the cycle.

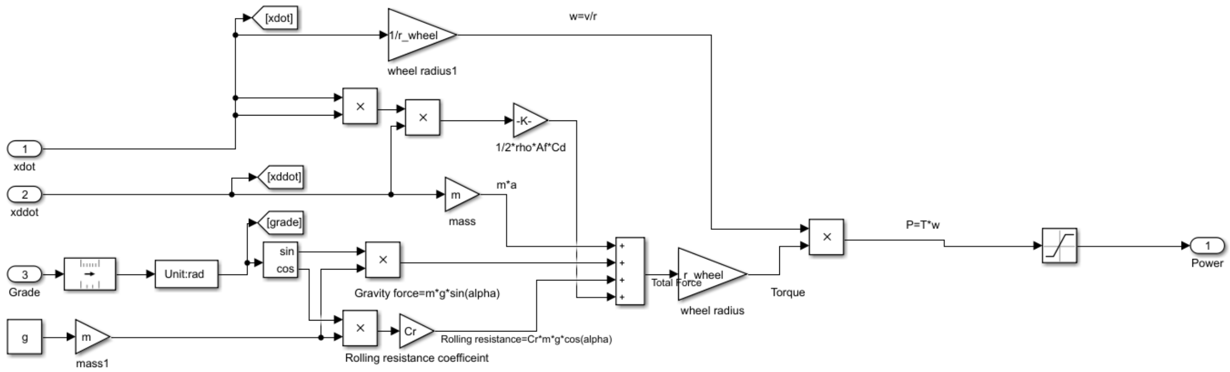


Figure 22 Summit model subsystem

The basic equation used to create the Summit model, was the second law of Newton, applied to a vehicle with losses caused by air resistance, rolling resistance and gravity:

$$\mathrm{F}_{total} = \mathrm{m} \cdot \mathrm{a} + \mathrm{F}_{air} + \mathrm{F}_{rolling} + \mathrm{F}_{gravity} \quad (2)$$

Where:

$$\mathrm{F}_{air} = \frac{1}{2} \cdot \rho \cdot Af \cdot Cd \cdot v^2 \quad (3)$$

$$\mathrm{F}_{rolling} = \mathrm{m} \cdot \mathrm{Cr} \cdot \mathrm{g} \cdot \cos\alpha \quad (4)$$

$$\mathrm{F}_{gravity} = \mathrm{m} \cdot \mathrm{g} \cdot \sin\alpha \quad (5)$$

Where m is the mass of the vehicle, a the acceleration, ρ the air density, Af the front vehicle area, Cd the drag coefficient, v the velocity, Cr is the rolling resistance coefficient and α is the angle of inclination of the road.

When the total force is calculated it is multiplied by the radius of the wheel (r_wheel), so we get the torque. Once the torque is obtained, it is multiplied by the rotational speed ($\omega = \frac{v}{r\_wheel}$) and the demanded power is obtained. At last, there is a saturation block, because in this project the regenerative breaking is not considered, so negative powers cannot be demanded.

Battery model

This model has been supplied by the DISA-US. In the mask of the model initialization the written command is a script called init_BateriaAeropack.m that will be added to the appendix codes. The following data must be introduced:

- Energy capacity of battery [Ah]
- Initial charge of battery [%]
- Current limit: minimum time to charge/discharge the battery [min]

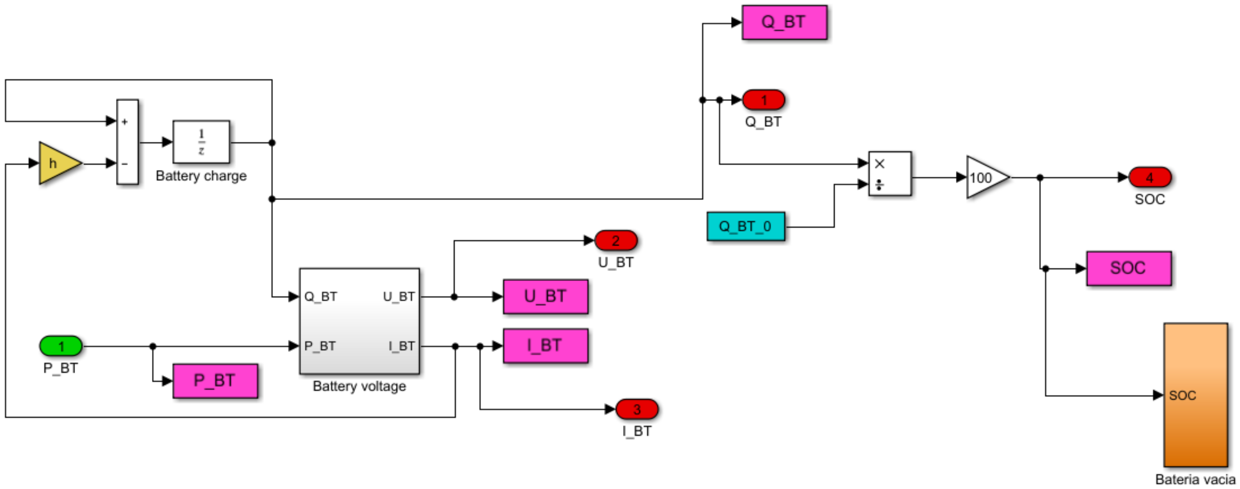


Figure 23 Submodel of the Li-ion battery

Fuel cell model

This model has also been supplied by the DISA-US. In the mask of the model initialization the written command is a script called init_PilaAeroDAQINTA.m that will be added to the appendix codes. The following data must be introduced:

- Number of fuel cells per stack (series connection) [-]
- Size of fuel cell [m^2]
- Theoretical fuel cell voltage [V]
- Idle power [W]

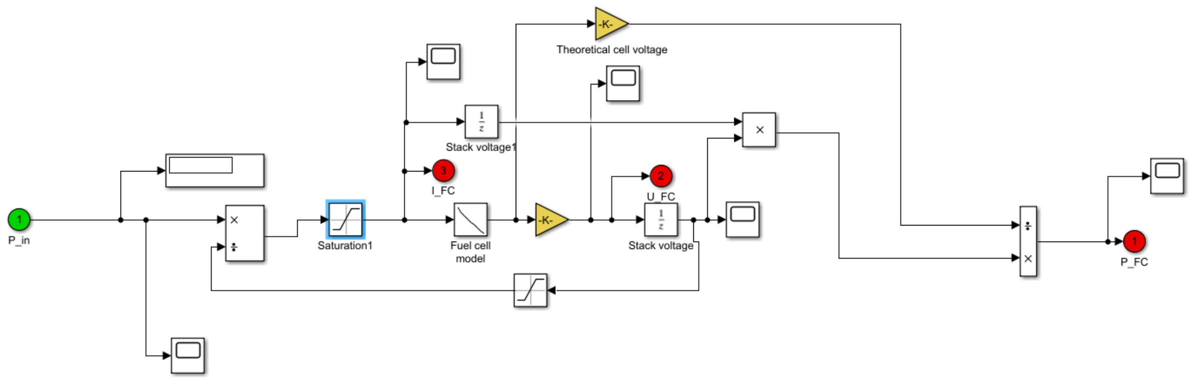


Figure 24 Fuel cell subsystem

The model is based on a lookup table to interpolate values of the curve of polarization of the fuel cell with twelve real points.

5 CONTROL STRATEGIES

HEURISTIC CONTROL

Heuristic strategy is based on intuitive rules and correlations involving various vehicular variables. For this project, the chosen guiding principle is that the battery discharge and charge phases should be regulated such that the SOC stays within predefined limits. Therefore, when the SOC drops below a certain level, the recharge mode should be favoured, while when the SOC exceeds a threshold the use of the fuel cell would be more appropriate. Furthermore, it is also advised that the fuel cell works in a constant power rate, so it is not constantly turning it on and off. [16]

So, as there is a fuel cell and another power source (Li-ion battery), the chosen working range for the fuel cell will be between 50 and 180 W.

Furthermore, the control usually should check if the power demand is positive or negative, because if there are negative power demands, that means that there is regenerative braking and the battery is going to be charged in this period, but it is not our case to study, since there is not regenerative braking.

Once it is checked that only positive power is being demanded by the vehicle, it is necessary to know if they are in the range of the fuel cell, below or above; and depending on the SOC the fuel cell will be set between the decided range, either for charging the battery or working at maximum performance with help from the battery.

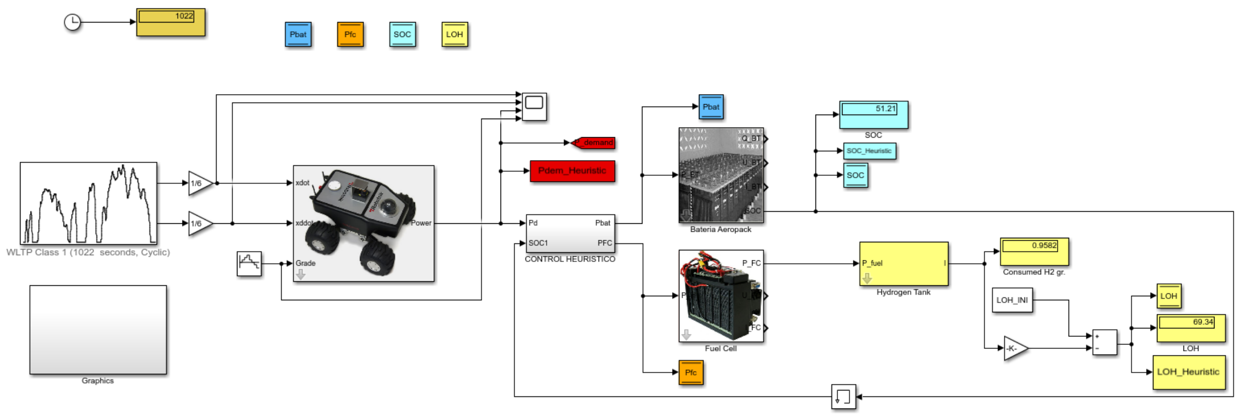


Figure 25 Heuristic control model

In my script of heuristic control, the first step is checking whether the SOC is between 40% and 75% or not, since it was one of the desired guiding principles:

<u>Regular SOC</u>

If the SOC is high enough the next step would be checking if the fuel cell is below, above or within its limits:

1. If the fuel cell is within its range, the power will be supplied entirely by the fuel cell. In case the SOC is below 50% the fuel cell will work at its maximum range (180W) and the extra power will be used to recharge the battery.
2. In case we are above the fuel cell range, two cases should be considered:
    a. If the battery is over 60% charged, the fuel cell will work at its maximum efficiency point and the battery will supply the rest.
    b. If the battery is below 60% charged, the fuel cell will work at its maximum performance point and the battery will supply the rest.
3. When the demanded power is below the range, two cases are considered again:

a. If the SOC is below 50% the fuel cell will work at its maximum efficiency point and the power surplus will recharge the battery.

b. If the SOC is above 50% the fuel cell will supply its minimum and the power surplus will recharge the battery.

Low SOC

If the state of charge were below 40% the code would check if the demanded power is below the nominal power that can be supplied by the fuel cell. The fuel cell will supply its nominal power, and if it is higher than the demand the surplus will recharge the battery. The decision of using the fuel cell above its working range was made because the limit values of the SOC have been considered more important in this project.

High SOC

On the other hand, if the SOC is over 75%, the battery will supply all de power demand and the fuel cell will be turned off. As explained before, it was decided to benefit the SOC within its limits rather than respect the fuel cell range**.**

At last, the output ports values are updated to obtain the power demanded to the battery and/or the fuel cell.

ECMS CONTROL

Equivalent consumption minimization strategy, also known as ECMS, is a strategy derived from Pontryagin's Minimum Principle and it is an online sub-optimal controller. Its main goal is the correct distribution of the power flows between the battery and the fuel cell, which are the sources supplying the demanded power, so the power flow is [16]:

$$P_{Demand}(\mathrm{t}) = P_{FC}(\mathrm{t}) + P_B(t) \quad (6)$$

where $P_{FC}$is the power produced by the fuel cell and $P_B$is the power produced by the battery.

The power flows are defined as positive when they come from the battery or the fuel cell, and negative when it is charging the battery.

The control variable will be the power distribution factor defined as $u(t) = \frac{P_B(t)}{P_{Demand}(\mathrm{t})}$. So, if u=0, it means that all the power comes from the fuel cell and if u=1 all de power comes from the battery (if its positive) or goes to the battery (if its negative).

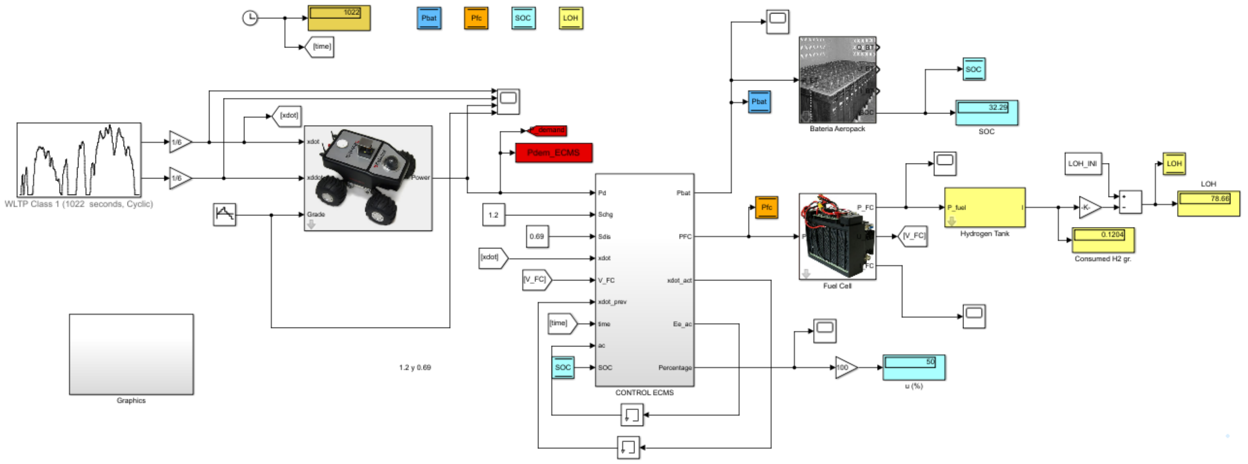


Figure 26 ECMS Control model

ECMS propose to replace the global minimum consumption for a local minimum consumption, so the problem is reduced to one time instant. For each time t with a time increment Δt, the ECMS control updates the control variable u(t) which minimizes the cost function defined below [7]:

$$J(\mathrm{t}) = P_{FC}(\mathrm{t}) + s(t) \cdot P_B(t) \quad (7)$$

Where P_{FC} and P_B are the power flow towards the fuel cell and the power flow toward or from the energy stored in the battery in the time interval Δt. The factor s(t) is the equivalent factor in which is based the ECMS control and it is used to ease the conversion of a electric flow in a chemical power flow.

The calculation of s(t) is the most important task in ECMS control. If s(t) is too big the use of energy from the battery will be penalized and the fuel cell consumption will rise. On the other hand, if the value is too small the battery could empty. That is why s(t) is calculated considering two possible values that depends on the E_{bat} and Δt considered [17]

$$s = \begin{cases} s_{dis}, & x > 0, \\ s_{chg}, & x < 0, \end{cases} \qquad E_B(t_f) = \int_0^{t_f} P_B(\tau)d\tau \tag{8}$$

The chosen procedure to evaluate s_{dis} *and* s_{chg} requires collecting data on the electrical energy use $E_B(t_f)$ and the fuel cell energy use $E_{FC}(t_f)$ over a mission of duration t_f. Various constant values are chosen in the range $u \in [-ul, ur]$ (Vehicle propulsion systems.).

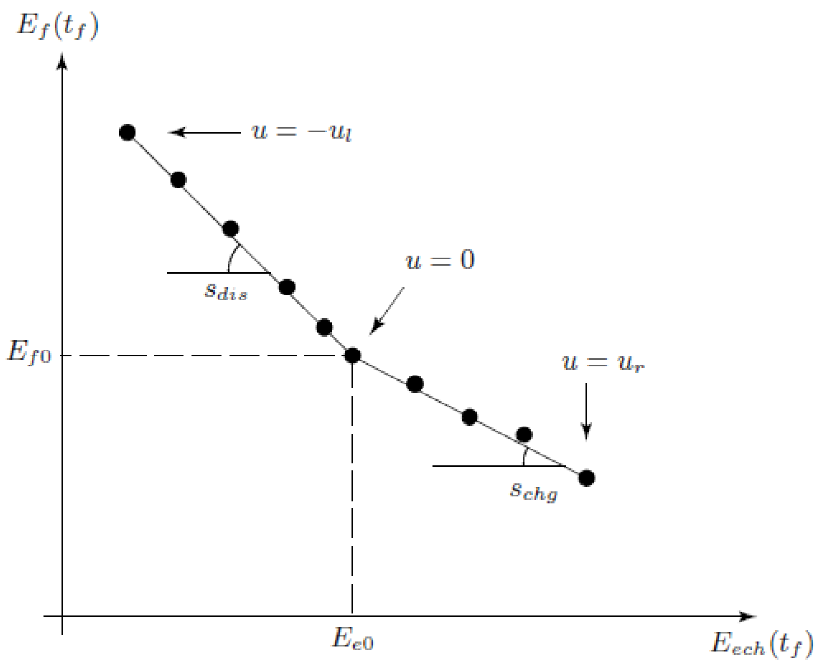


Figure 27 Typical dependency between Efc and Ebat [16]

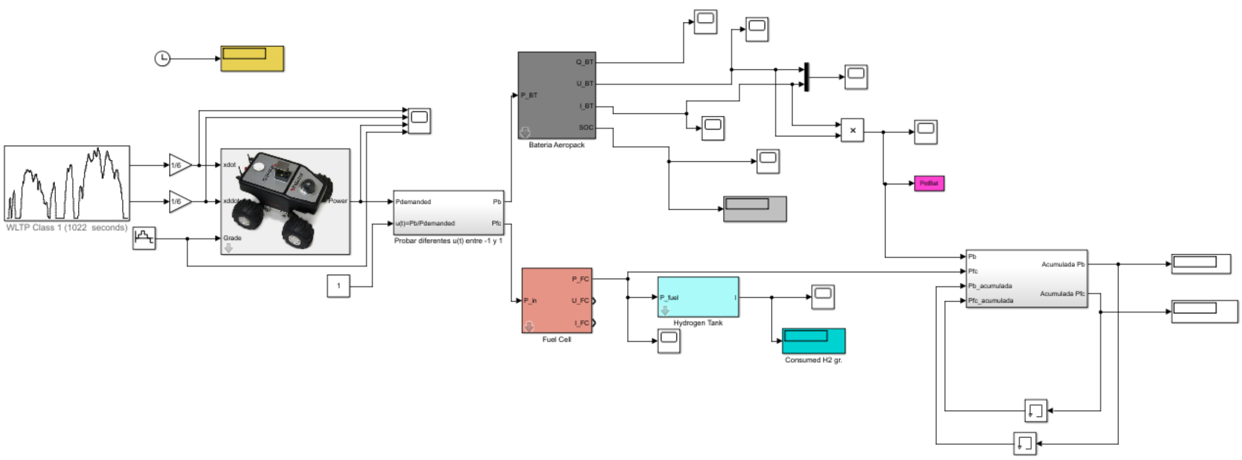


Figure 28 Model to calculate the constant equivalent factors.

From this last model, the values of the constant equivalent factors are calculated thanks to the following graphics:

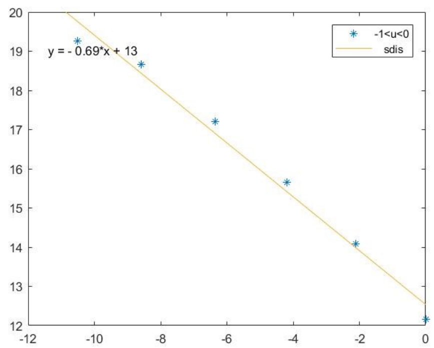


Figure 29 s_{dis}

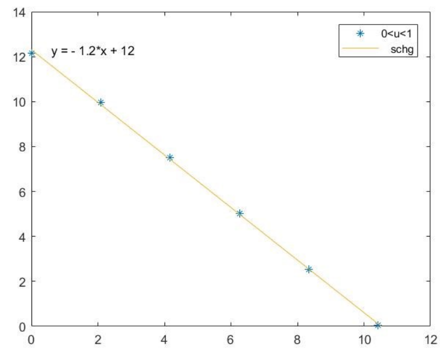


Figure 30 s_{chg}

Once s_{dis} and s_{chg} are obtained the next step to calculate s(t) is to calculate the probability p(t):

$$P(\mathrm{t}) = \frac{E_B(t) + E_{dem}^h}{E_{FC}^h} \tag{9}$$

Given P_{FC}^{max}, the average power demanded can be easily calculated:

$$P(\mathrm{t}) = \frac{E_B(t)}{t_h \cdot P_{FC}^{max}} + \frac{P_{dem}^{avg}}{P_{FC}^{max}} \tag{10}$$

Where $E_B(t)$ is the accumulated energy of the battery:

$$E_B(t) = E_{B\_accumulated} + (P_B + \frac{1}{2} \cdot m \cdot (v_{prev}^2 - v_{act}^2)) \tag{11}$$

Finally, s(t) will be obtained as follows:

$$s(\mathrm{t}) = s_{dis} \cdot P(t) + s_{chg} \cdot (1 - P(t)) \tag{12}$$

So, if p(t)=1 it means that s(t)=s_{dis} and if p(t)=0 means s(t)=s_{chg}.

Once the equivalent factor is obtained, it is necessary to calculate the efficiency of the fuel cell, interpolating in the graphic of the fuel cell manufacturer [7].

At last, the value of the cost function is calculated, and if it is lower than J_{min}, this value is updated and its correspondent u(t) is stored:

$$\mathrm{J} = \frac{P_{FC}(t)}{\eta_{FC}(t)} + s(t) \cdot P_B(t) \tag{13}$$

MODEL PREDICTIVE CONTROL

MPC is an optimization-based method that can compute control actions in order to fulfil some criteria. In this sense, it is similar to any other optimization-based strategy.

But the main advantage of MPC is that the optimization process is embedded in a control scheme which incorporates feedback.

This way, MPC can face disturbances and model mismatch, recomputing the necessary control actions in a receding way when fresh information about the system state is available.

The main feature of MPC is the replacement of an (usually complex) off-line determination of the control actions by a repeated on-line solution of the optimization of an open-loop problem whose solution provides the current control action. [18]

5.3.1 Control-oriented Model

MPC needs a model of the system to perform predictions. This control-oriented model is a simplified one that can be integrated in the optimization procedure.

the main dynamics to be considered is that of the storage units, which, together with the balance equation of powers in the bus, will constitute the model to be used by MPC.

In the case of storage systems, which can inject or extract from the bus, their power is considered positive when discharging and negative when charging.

The stored units can be modelled by an energy balance equation that determines the increment in the level of energy x(t) as the integral of the charged power $P_{sto}(t)$ which is positive for charging and negative for discharging:

$$x(t+1) = x(t) - \eta\, T_s P_{sto}(t) \tag{14}$$

where Ts is the sampling time, given in seconds. In general, the influence of the charge/discharge of the storage units on the stored energy levels is not the same, so different efficiencies for charge/discharge are used. In order to manage the different behaviour in charging and discharging, a binary variable $\delta(t)$ must be considered, which takes value 1 for charging and 0 for discharging. this equation is nonlinear and includes continuous and binary variables (it is a hybrid model), therefore is not easy to manage. The problem can be simplified if the different efficiencies for charge/discharge are neglected, considering η in equation (14) as a fixed value. [18]

In the common case that the storage unit is a battery, the level of energy is given by the SOC, defined as the ratio between the current capacity $C_{bat}(t)$ and the maximum battery capacity C_{max} then its evolution is given by:

$$SOC(t+1) = SOC(t) - \frac{\eta_{bat}\, T_s}{C_{max}} P_{bat}(t) \tag{15}$$

In this project there is also a fuel cell, where the level of hydrogen in the hydrogen tank is given by the LOH (Level of Hydrogen), and its evolution is given by:

$$\mathrm{LOH}(t+1) = \mathrm{LOH}(t) - \frac{T_s}{\eta_{fc} V_{max}} P_{fc}(t) \tag{16}$$

The mean value obtained for the conversion coefficient of the battery was $K_{bat} = \frac{\eta_{bat}}{C_{max}} = \frac{0.85}{260*6*0.38}? = 1.15 \cdot 10^{-3}\ \frac{\%}{kW\ s}$. In the case of the fuel cell the mean value was $K_{fc} = \frac{1}{\eta_{fc} V_{max}} = \frac{1}{0.6 \cdot 0.1} = 16.67\ \frac{\%}{kW\ s}$. [18]

Then the model of the system is formed by one equation per storage unit and the following energy balance equation of the energy system, which implies that the net sum of all the energy flows in the bus is zero:

$$\sum_{i=1}^{n_b} P_{bat,i}(t) + \sum_{i=1}^{n_{fc}} P_{fc,i}(t) - \sum_{i=1}^{n_l} P_{load,i}(t) = 0 \tag{17}$$

5.3.2 State space model

The states x(t) are the energy stored in the different energy storage systems: SOC of battery and Level of Hydrogen (LOH) in tanks. Usually, the outputs y(t) will coincide with the states and the manipulated variables u(t) will be the power flows that can be manipulated to charge or discharge the battery and power supply by the fuel cell. So, the following vectors can be defined:

$$x(t) = [SOC(t)\ \mathrm{LOH}(t)]^T \tag{18}$$

$$u(t) = [P_{fc}(t)\ P_{bat}(t)]^T \tag{19}$$

$$d(t) = \sum_{i=1}^{n} P_{gen,i}(t) - \sum_{i=1}^{n_l} P_{dem,i}(t) \tag{20}$$

$$y(t) = x(t) \tag{21}$$

and the dynamics can be written in the general state space form with appropriate matrices:

$$\begin{aligned} x(t+1) &= \mathrm{A}x(t) + \mathrm{B}u(t) + B_d d(t) \\ y(t) &= \mathrm{C}x(t) \end{aligned} \tag{22}$$

Where A and C are equal to the identity matrix I and B and B_d are composed of terms that depend on the storage efficiency, which is used to convert the input/output flows of a storage device into its stored energy.

$$A = I, \quad B = \begin{bmatrix} K_{bat} & K_{bat} \\ -K_{fc} & 0 \end{bmatrix}, \quad B_d \begin{bmatrix} K_{bat} \\ 0 \end{bmatrix}, \quad C = I \tag{23}$$

5.3.3 Controller design

The formulation of the MPC problem requires the definition of the cost function to be minimized and operational constraints to be imposed.

5.3.3.1 Cost function

The goal of this multiobjective optimization problem is to accomplish an optimal solution for several objectives, so the result will be a compromise among the objectives. Consequently, the solution will be a state where no objective can be improved without sacrificing at least another. [18] [19]

The cost function can include terms that consider the values of the different powers involved and the power rates. It can also penalize the deviation of the stored energy from a desired operation point. A cost function can be customized to this case study:

$$\begin{aligned} J = \sum_{k=1}^{N_c} \alpha 1\, P_{fc}^2(t+k) + \alpha 2\, P_{bat}^2(t+k) + \beta 1\, \Delta P_{fc}^2(t+k) + \beta 2\, \Delta P_{bat}^2(t+k) \\ + \sum_{k=1}^{N_p} \gamma 1\, (SOC(t+k) - SOC_{ref})^2 + \gamma 2(\, LOH(t+k) - LOH_{ref})^2 \end{aligned} \tag{24}$$

Setpoint tracking is a substantial issue, high values ($\gamma_1 = 10^{-6}$ and $\gamma_2 = 10^{-6}$) have been chosen for their associates weights. The other ones are: $\alpha_1 = 10^{-4}$, $\alpha_2 = 10^{-4}$, $\beta_1 = 0.01$, $\beta_2 = 0$. The choice of these weights encourages the use of hydrogen versus the battery. The chosen horizons are Np = 5 and Nc =10.

5.3.3.2 Constraints

There are basically two types of constraints: those associated to physical limits of the units that cannot be trespassed and those related to operational limits that should not be exceeded. The first type includes the limited power that can be supplied by the units. Those are physical thresholds that cannot be trespassed for constructive reasons which take the form:

$$\begin{aligned} &P_{bat}^{min} \le P_{bat}(t) \le P_{bat}^{max} && \forall t \\ &P_{fc}^{min} \le P_{fc}(t) \le P_{fc}^{max} && \forall t \\ &SOC^{min} \le SOC(t) \le SOC^{max} && \forall t \\ &LOH^{min} \le LOH(t) \le LOH^{max} && \forall t \end{aligned}$$

Notice that the maximum and minimum values can be exactly the physical limits, but a safety band can also be considered, avoiding working very close to dangerous regions.

The second type of constraints are imposed to avoid sudden changes in the power supplied by the units. These are limits which affect the degradation of the units and will be important in expensive equipment such as fuel cells. [18]

$$\Delta P_{bat}^{min} \leq \Delta P_{bat}(t) \leq \Delta P_{bat}^{max} \qquad \forall t$$

$$\Delta P_{fc}^{min} \leq \Delta P_{fc}(t) \leq \Delta P_{fc}^{max} \qquad \forall t$$

$$\Delta SOC^{min} \leq \Delta SOC(t) \leq \Delta SOC^{max} \qquad \forall t$$

$$\Delta LOH^{min} \leq \Delta LOH(t) \leq \Delta LOH^{max} \qquad \forall t$$

Notice that some of these constraints can be moved to the category of soft constraints if the inequalities are substituted by a weighted term in the cost function.

These constraints can be quantified as shown in **Table 9**. Note that some of them are physical limits (e.g. power supplied by the fuel cell) while others are limits imposed for a safe operation (e.g. power rate requested to the fuel cell).

Table 9 Constraints [18]

	Power(W)	Power rate (W/s)	State of Charge (%)
Battery	0-250	Unconstrained	40-75
Fuel cell	0-200	20	-
H_2 storage	-	-	10-90

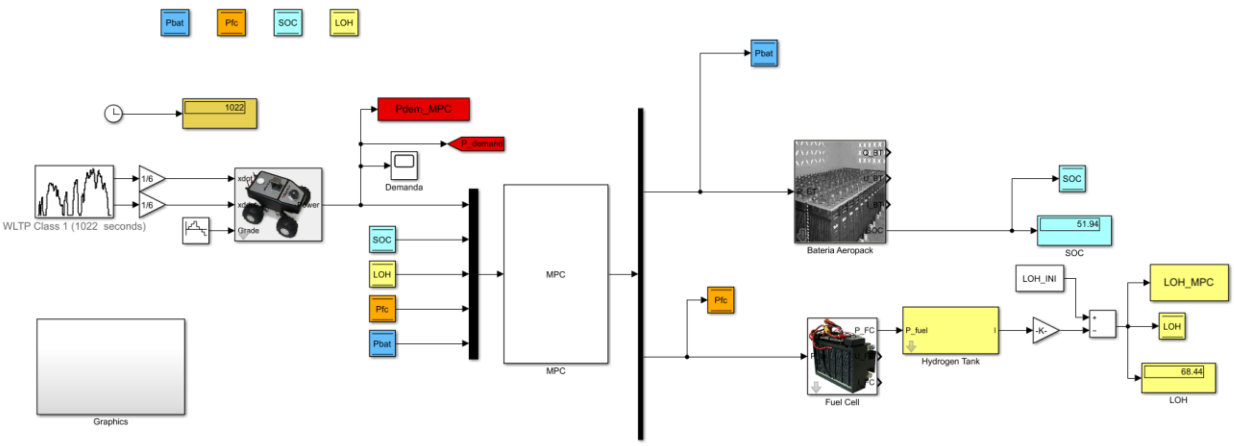


Figure 31 MPC model

6 CONTROL STRATEGIES COMPARISON

WLTP drive cycle

Under conditions defined by EU law, the Worldwide Harmonised Light Vehicle Test Procedure (WLTP) laboratory test is used to measure fuel consumption and CO2 emissions from passenger cars, as well as their pollutant emissions. The WLTP cycle was developed using real-driving data, gathered from around the world. The WLTP driving cycle is divided into four parts with different average speeds: low, medium, high and extra high [20].

These speeds have been modified in this project to adjust the maximum speed of the mobile robot, that is 3 m/s.

In addition, an inclination profile has been implemented for all the WLTP simulations.

6.1.1 Heuristic

As it is shown in **Figure 32** the heuristic controller works as desired, so the fuel cell gives 180 W when needed, and the rest is provided by the battery. In these conditions, the fuel cell gives most of the power, because the LOH is high and is desired to charge the battery. When the demand is less than 50 W, the fuel cell provides 50 W and the excess is used to charge the battery.

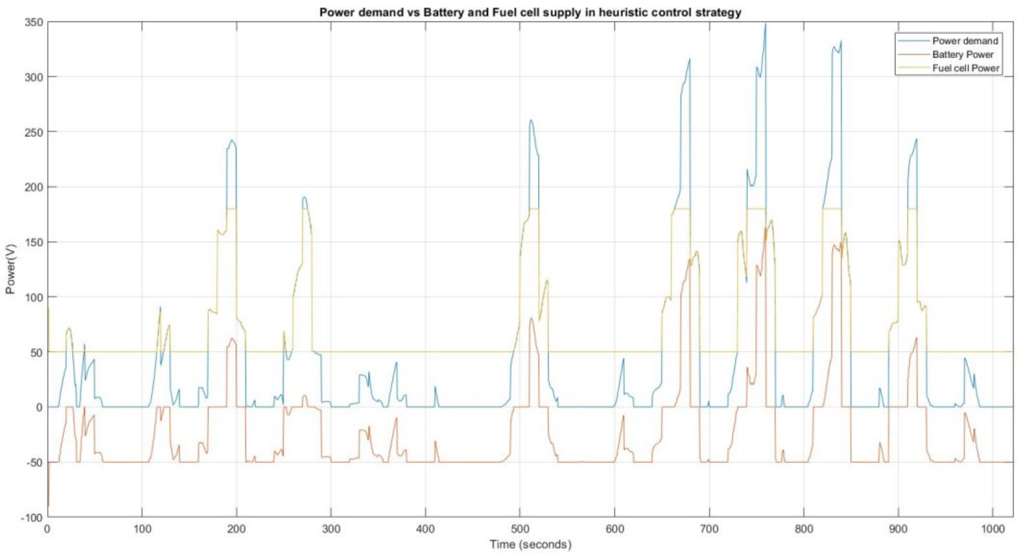


Figure 32 Power in heuristic strategy with WLTP cycle

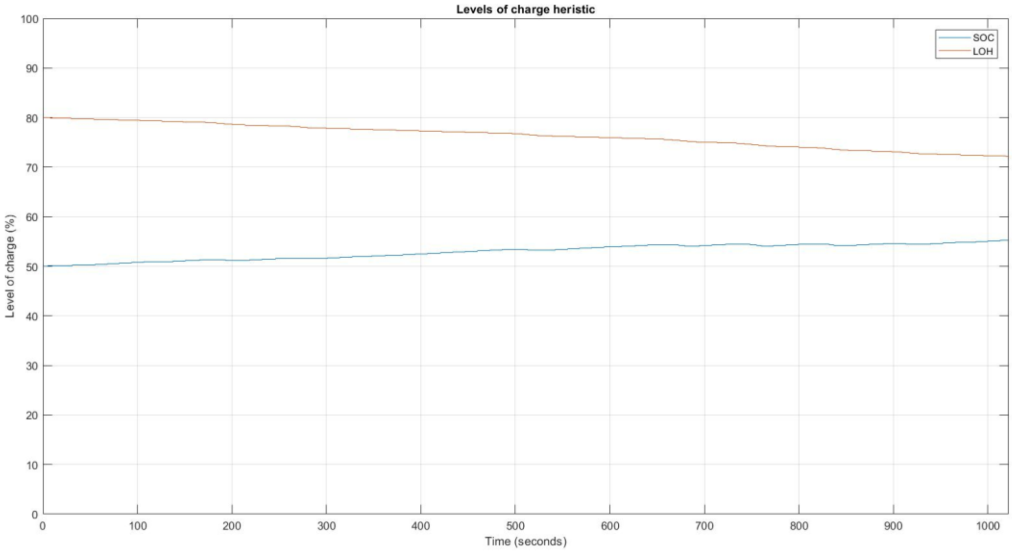


Figure 33 Levels of charge in heuristic strategy with WLTP cycle

6.1.2 ECMS

With the ECMS control strategy with WLTP cycle it is very clear that most of the time the best option to deliver the power is the battery, while the fuel cell only helps when the demand is above 250 W. That's why the LOH barely decreases and the SOC decreases about 10% in this simulation.

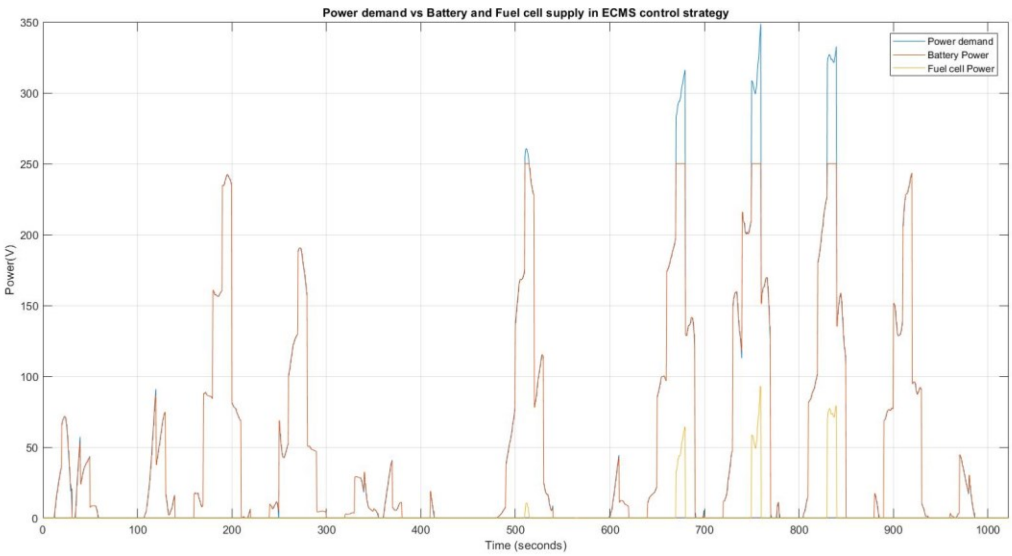


Figure 34 Power in ECMS strategy with WLTP cycle

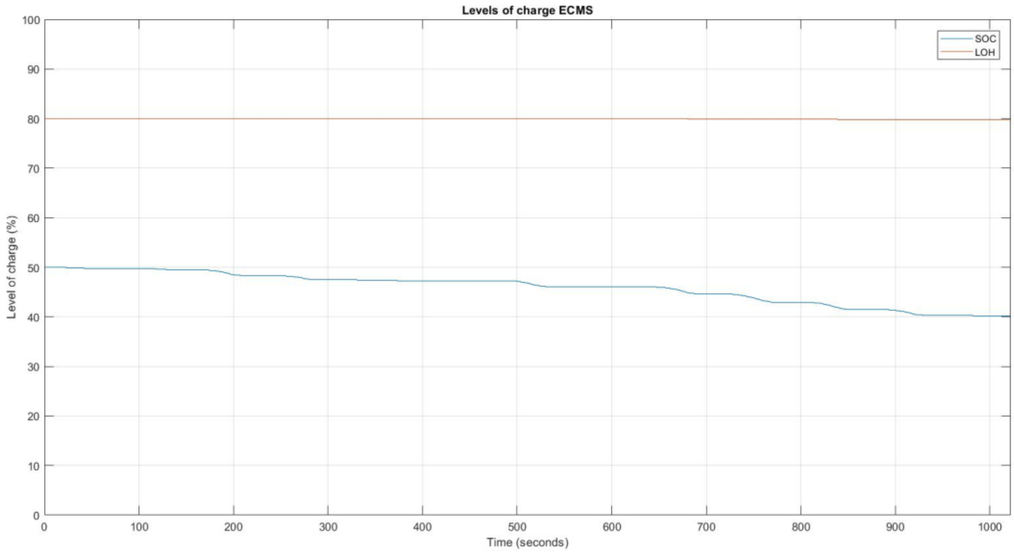


Figure 35 Levels of charge in ECMS strategy with WLTP cycle

6.1.3 MPC

With the MPC strategy, we can observe that in the beginning the fuel cell gives its maximum of 200W and the battery is charging until the SOC approaches 60%, when the power of the fuel cell starts to decrease, but is still high and most of the time keeps charging the battery.

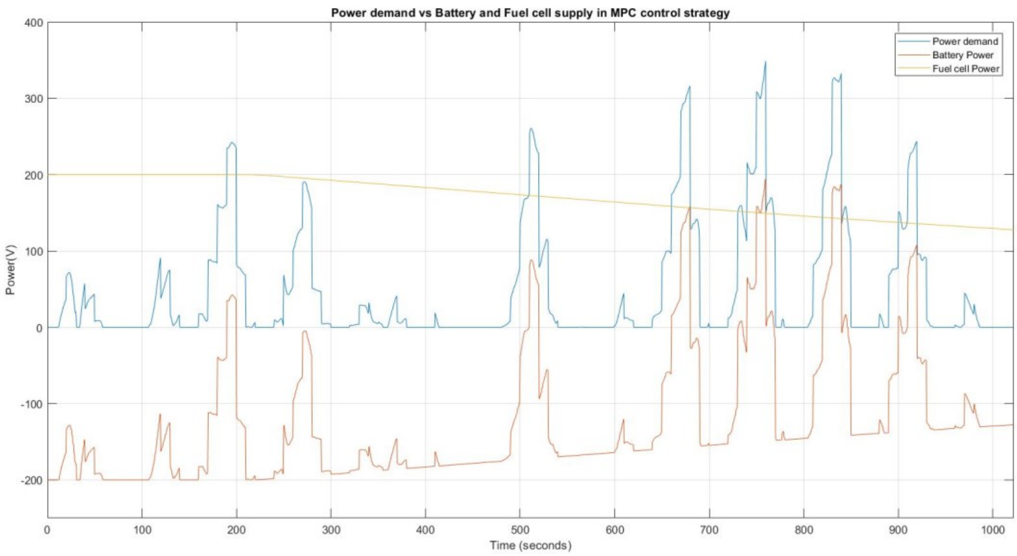


Figure 36 Power in MPC strategy with WLTP cycle

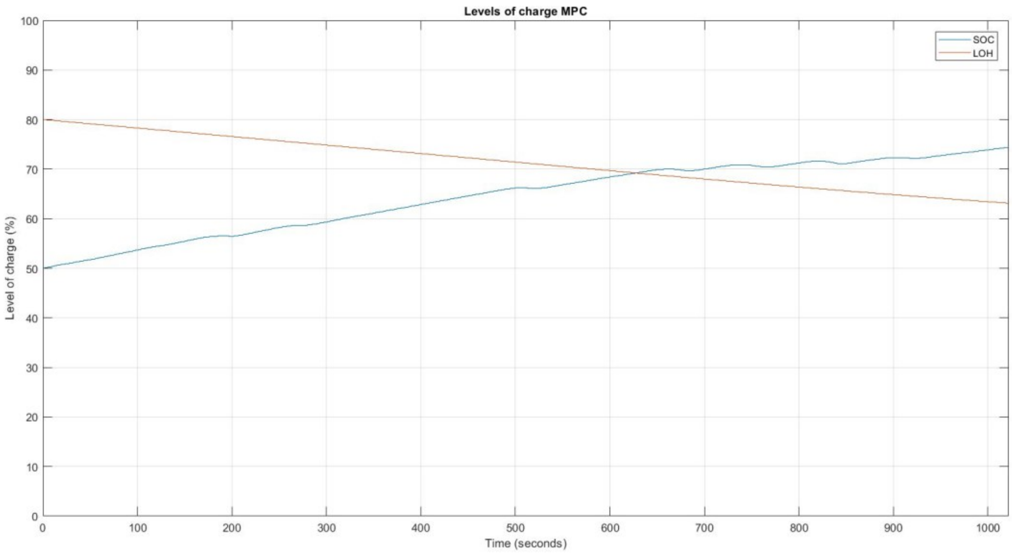


Figure 37 Levels of charge in MPC strategy with WLTP cycle

Table 10 Comparison of consumptions with WLTP cycle

	CONTROL STRATEGY	ΔSOC (%)	ΔLOH (%)	ENERGY (Wh)[1]
WLTP	Heuristic	-0.0143	-12.9534	16.95
	ECMS	-10.6307	-7.6928	13.59
	MPC	24.4459	-16.8832	17.38

In the WLTP simulations, we can realize that the ECMS strategy is the best in terms of energy consumption, while the MPC priority is keeping the SOC and the LOH within its limits as it avoids the degradation of the components limiting the power rate of the fuel cell. It is also remarkable that heuristic strategy has also a good behaviour, the consumption is a bit lower than MPC and the SOC and LOH are within the limits. The bigger problem in heuristic control in this case would be the power rates in the fuel cell that could cause degradation.

[1] The energy in this table, has been obtained integrating in the simulink model the sum of the powers consumed by the battery and the fuel cell and then multiplying it by a gain of 1/3600 so the result unit is in Wh instead of Ws

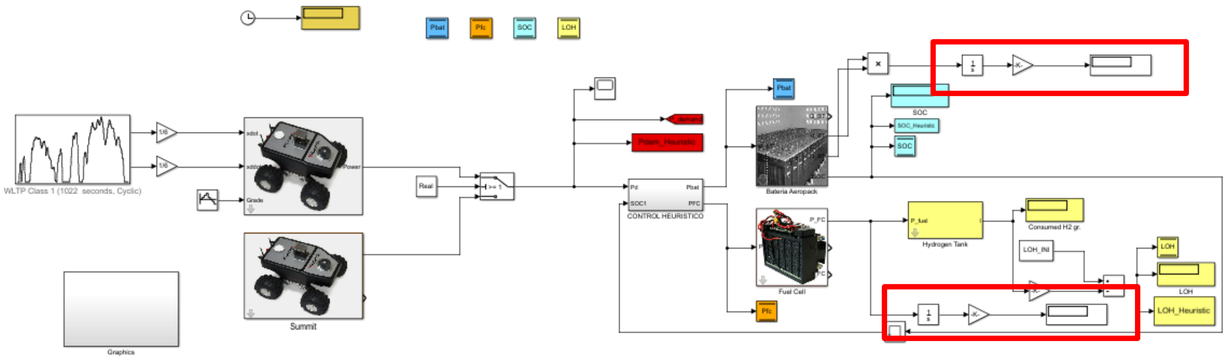


Figure 38 Energy consumption of the battery and the fuel cell in Wh

Mission 1

The simulations below have been made basing on real data acquired form the Summit XL power demand in a real mission. The mission duration has been set to 1022 seconds, so it will be easier to compare with the WLTP driving cycle, which has that duration.

6.2.1 Heuristic

In this case, we can observe that the fuel cell power varies between 50 and 180W while the battery helps delivering the rest of the power when the demand is above 180W and it recharges when the demand is below 50W. As shown in **Figure 40** the LOH decreases about 11.5% while the SOC variation is barely 0.3%, so it keeps around 50%.

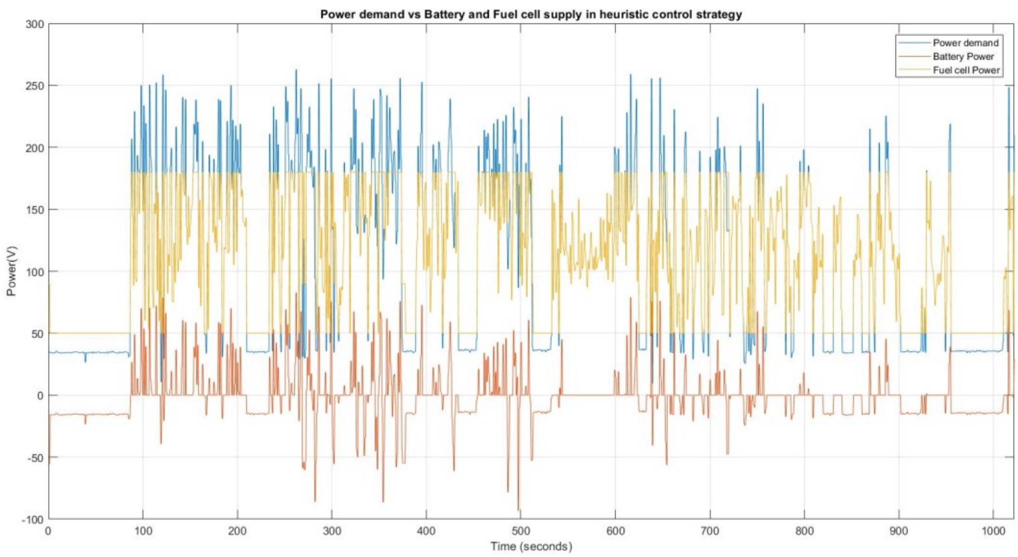


Figure 39 Power in heuristic strategy for real mission 1

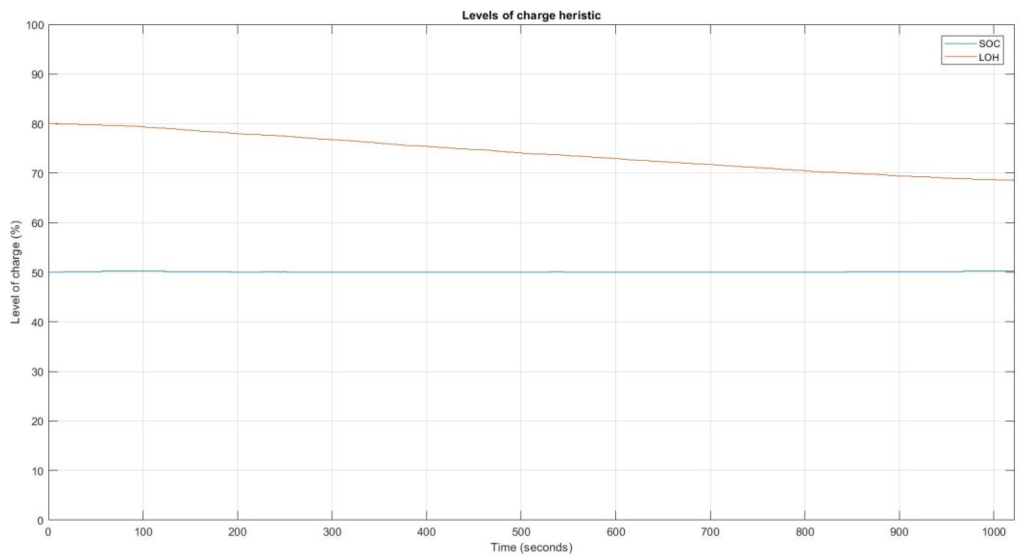


Figure 40 Levels of charge in heuristic strategy for real mission 1

6.2.2 ECMS

In the real mission 1 with ECMS control strategy, we can notice a curious behaviour. While the SOC is above 40%, all the power is delivered by the battery, except when the demand is above 250W and the fuel cell delivers the rest. However, when the SOC reaches 40% the dynamic changes and the fuel cell is the one giving its maximum power, while the battery only helps when the demand is above 200W.

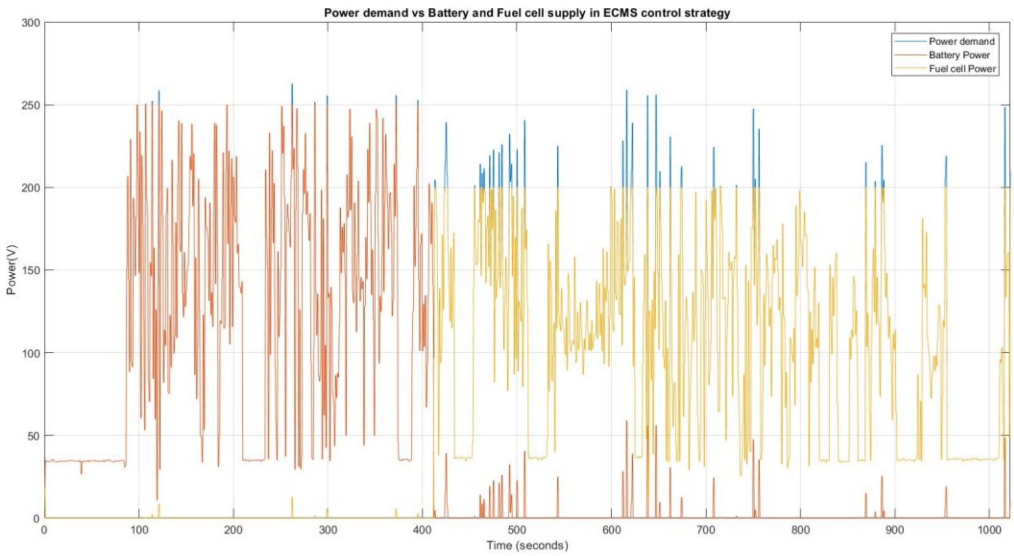


Figure 41 Power in ECMS strategy for real mission 1

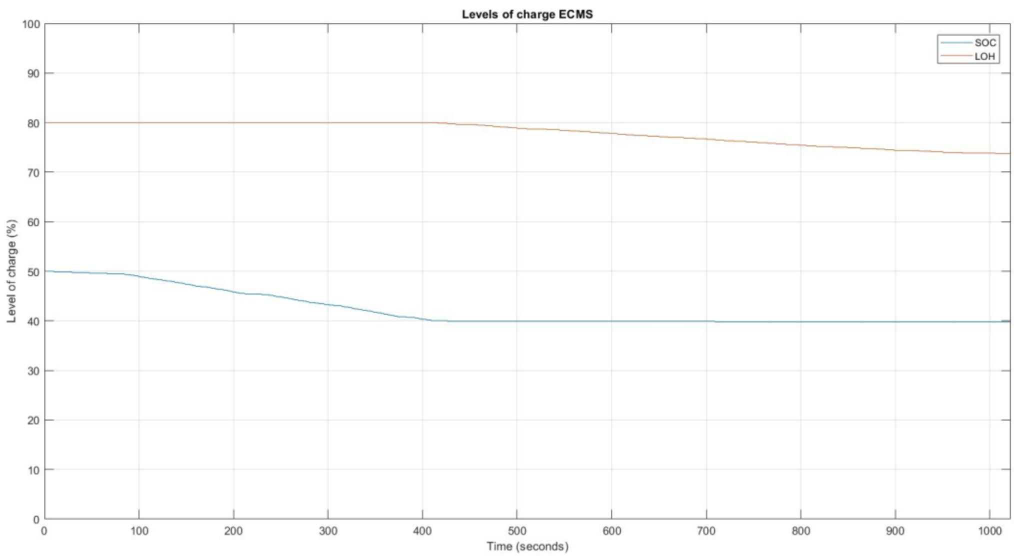


Figure 42 Levels of charge in ECMS strategy for real mission 1

6.2.3 MPC

When the MPC is used in a real mission, we get a quasi-optimal behaviour. The battery charges unless the power demand is higher than 200W and until the SOC is below 60%. Once the SOC reaches 60% the fuel cell power starts to decrease slowly, and the battery gives the rest of the power necessary to reach the demand.

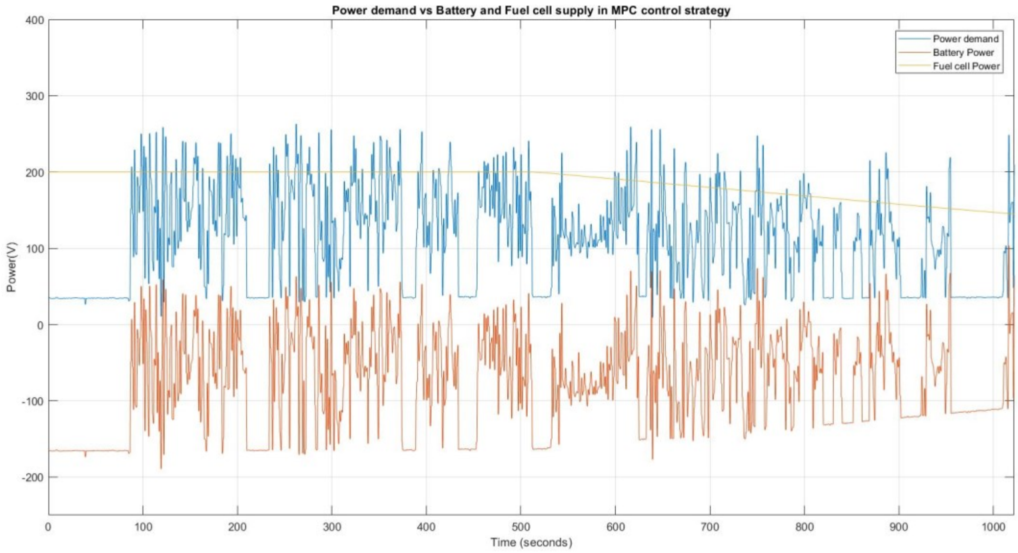


Figure 43 Power in MPC strategy for real mission 1

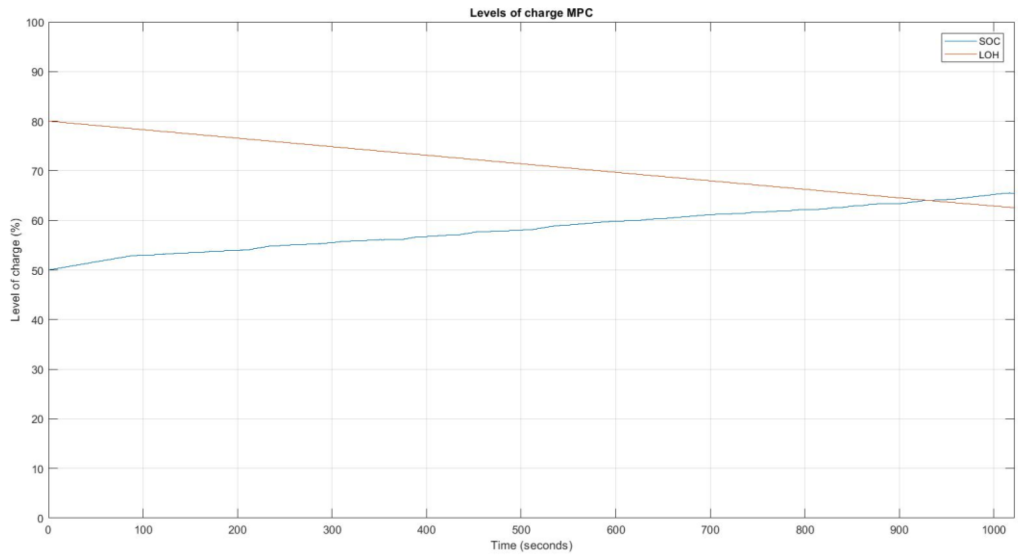


Figure 44 Levels of charge in MPC strategy for real mission 1

Looking at the comparison made in **Table 11** it is obvious that the heuristic controller has the worst performance in terms of energy consumption, followed by the ECMS strategy and the best one is the MPC strategy.

Regarding the levels of charge in the battery and the hydrogen tank, the three of them met the requirements.

Finally, limit imposed for a safe operation (power rate requested to the fuel cell) is only fulfilled in MPC controller. So, it is clearly the best controller to implement in the real robot, despite its higher computational charge.

Table 11 Comparison of consumptions for real mission 1

	CONTROL STRATEGY	ΔSOC (%)	ΔLOH (%)	ENERGY (Wh)[2]
MISSION 1	Heuristic	0.3124	-11.4999	35.28
	ECMS	-10.2019	-6.3546	33.22
	MPC	15.5348	-17.4432	32.26

[2] The energy in this table, has been obtained integrating in the simulink model the sum of the powers consumed by the battery and the fuel cell and then multiplying it by a gain of 1/3600 so the result unit is in Wh instead of Ws

Mission 2

As in the previous case, the simulations below have been made basing on real data acquired form the Summit XL power demand in a real mission. The mission duration has been set to 1022 seconds, so it will be easier to compare with the WLTP driving cycle, which has that duration.

6.3.1 Heuristic

This case is almost identical to the heuristic control of mission 1. We can observe that the fuel cell power varies between 50 and 180W while the battery helps delivering the rest of the power when the demand is above 180W and it recharges when the demand is below the fuel cell power. As shown in **Figure 46** the LOH decreases about 13% while the SOC variation is barely null, so it keeps around 50%.

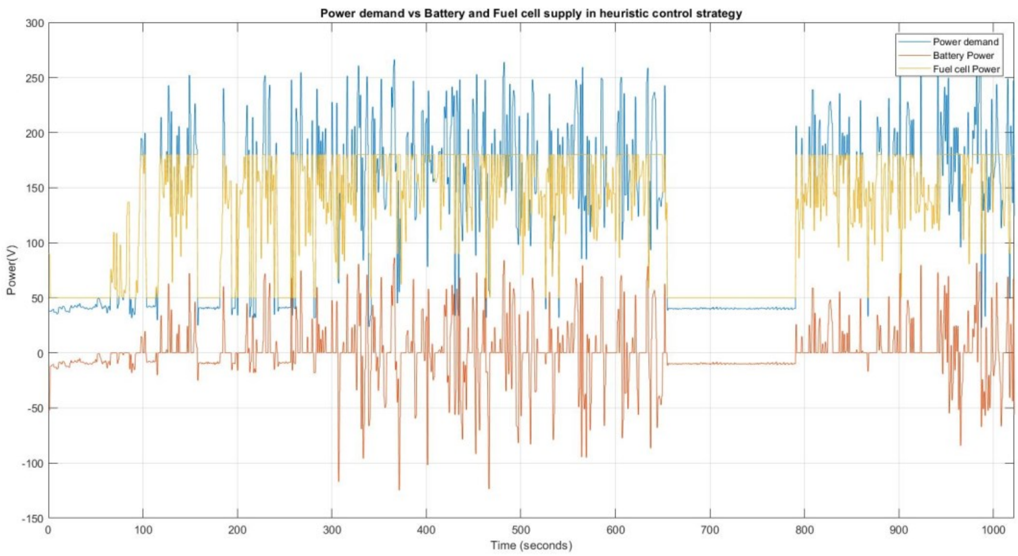


Figure 45 Power in heuristic strategy for real mission 2

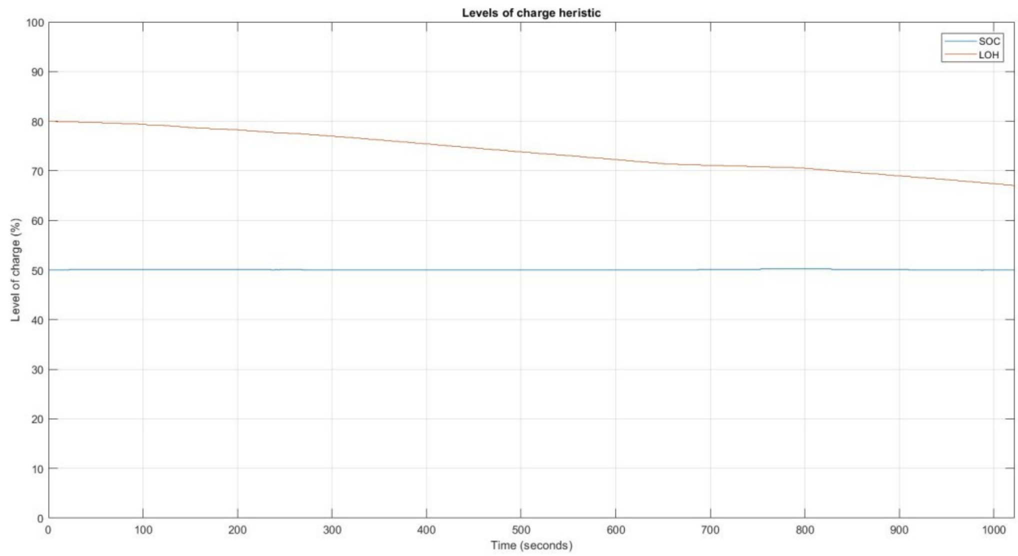


Figure 46 Levels of charge in heuristic strategy for real mission 2

6.3.2 ECMS

As we noticed in real mission 1 with ECMS control strategy (**Figure 41** and **Figure 42**), while the SOC is above 40%, all the power is delivered by the battery, except when the demand is above 250W and the fuel cell delivers the rest. However, when the SOC reaches 40% the dynamic changes and the fuel cell is the one giving its maximum power, while the battery only helps when the demand is above 200W. The same behaviour can be appreciated in real mission 2 with ECMS control strategy (**Figure 47** and **Figure 48**).

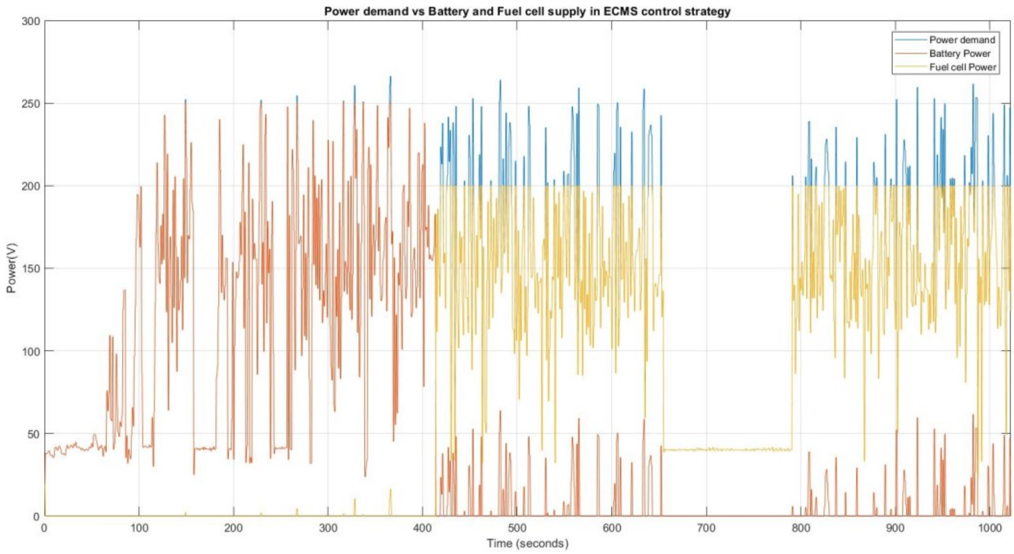


Figure 47 Power in ECMS strategy for real mission 2

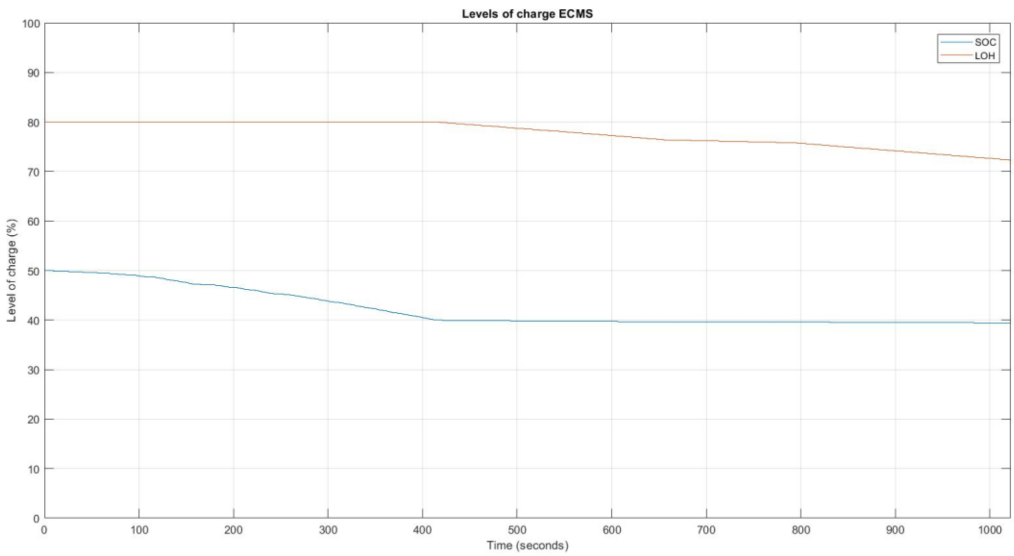


Figure 48 Levels of charge in ECMS strategy for real mission 2

6.3.3 MPC

In this last case of study, MPC control in real mission 2 we can appreciate a similar behaviour with the mission 1 MPC, and we also get a quasi-optimal solution, so it can make us believe that it really is the best of the three proposed control strategies for the energy management system of the Summit mobile robot.

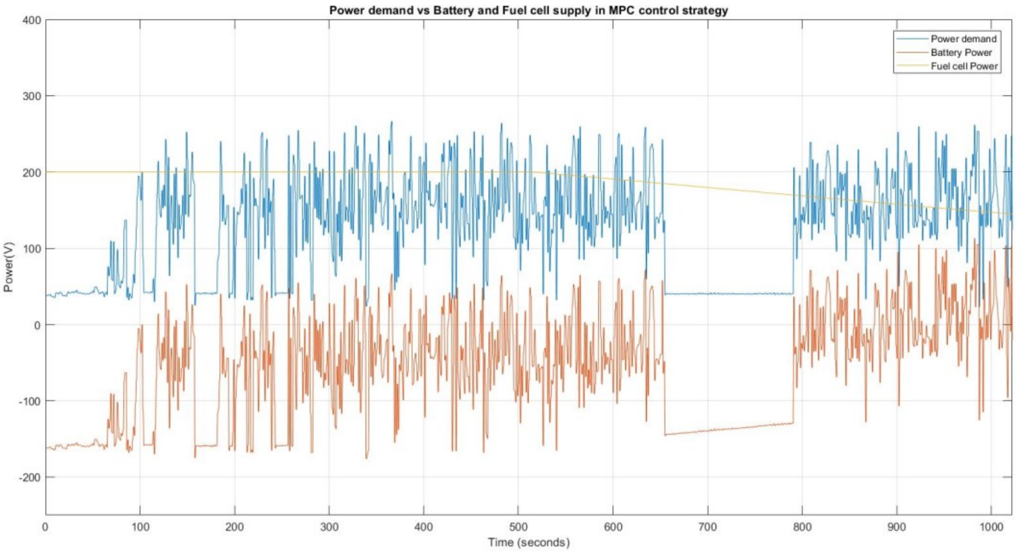


Figure 49 Power in MPC strategy for real mission 2

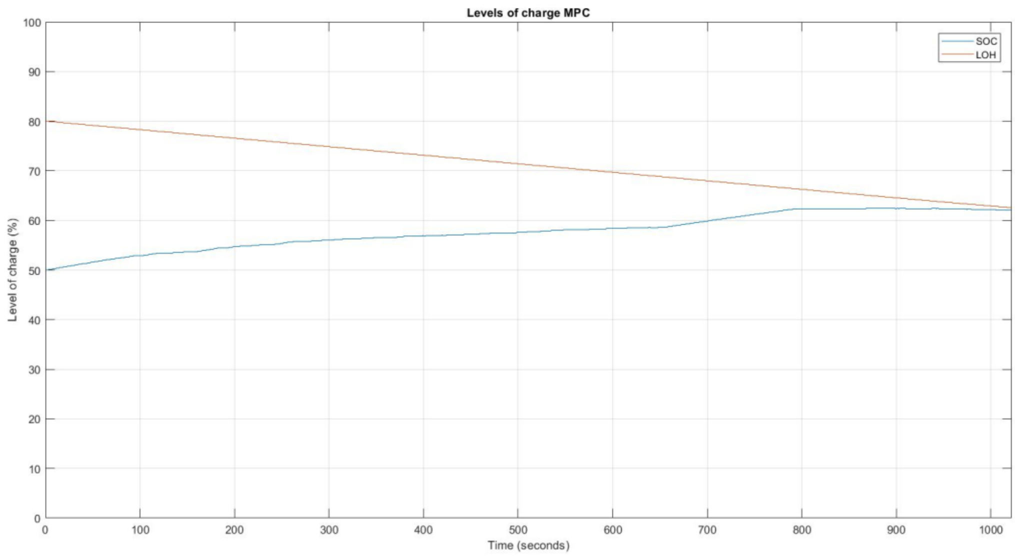


Figure 50 Levels of charge in MPC strategy for real mission 2

Table 12 Comparison of consumptions for real mission 2

	CONTROL STRATEGY	ΔSOC (%)	ΔLOH (%)	ENERGY (Wh)[3]
MISSION 2	Heuristic	-0.0143	-12.9534	40.2
	ECMS	-10.6307	-7.6928	37.96
	MPC	12.0675	-17.4432	37.18

[3] The energy in this table, has been obtained integrating in the simulink model the sum of the powers consumed by the battery and the fuel cell and then multiplying it by a gain of 1/3600 so the result unit is in Wh instead of Ws

Overall comparison between the three control strategies

As previously discussed in this project, the best performance for real mission is obtained with the MPC controller, but the cost is a higher computational cost.

None of the controllers has a bad behaviour in terms of energy consumption. The heuristic control is clearly the worst, but it is the easiest one to implement and the one with less computational cost.

With the ECMS control we can obtain very good results in terms of energy consumption and high efficiency, but it does not have the same behaviour as the MPC in terms of battery and fuel cell care.

The results are coherent with the graphics we have seen for the three cases (WLTP, M1 and M2). The heuristic controller always follows the same simple rules and have a decent performance. The ECMS controller priority is to use the battery, instead of the fuel cell, but when the SOC reaches its minimum, the fuel cell takes the leading role.

Finally, the MPC follows the levels of charge references for the battery and the fuel cell, even the weight of the reference following was set very low. It takes the levels of charge within their limits, respects the maximum and minimum powers for the battery and the fuel cell and also respect the limit in the fuel cell power rate to protect the fuel cell.

Table 13 Control strategies consumptions comparison

	CONTROL STRATEGY	ΔSOC (%)	ΔLOH (%)	ENERGY (Wh)[4]
WLTP	Heuristic	-0.0143	-12.9534	16.95
	ECMS	-10.6307	-7.6928	13.59
	MPC	24.4459	-16.8832	17.38
MISSION 1	Heuristic	0.3124	-11.4999	35.28
	ECMS	-10.2019	-6.3546	33.22
	MPC	15.5348	-17.4432	32.26
MISSION 2	Heuristic	-0.0143	-12.9534	40.2
	ECMS	-10.6307	-7.6928	37.96
	MPC	12.0675	-17.4432	37.18

[4] The energy in this table, has been obtained integrating in the simulink model the sum of the powers consumed by the battery and the fuel cell and then multiplying it by a gain of 1/3600 so the result unit is in Wh instead of Ws

7 CONCLUSIONS

The main goal of this project was to make a model of the hybrid mobile robot with its energy management system and propose three different strategies to control the battery and the fuel cell.

Using mathematical and physical approaches, a complete model of the Summit XL vehicle was obtained. Combining it with the model of the battery and the fuel cell provided by the DISA-US a complete model of the energy management system the final model was developed.

The three control strategies were heuristic, equivalent consumption minimization strategy and model predictive control.

Comparing those controls, the conclusion is that the MPC has the best performance, followed by the ECMS and finally the heuristic one.

Future lines of research should improve the results of the controllers. First, the models could be improved to get more accurate results for the simulations, and lots of test should be made to ensure that the controllers, specially the ECMS and the MPC, are as well as they can possibly be. It is also important to run test with different initial conditions and with different missions and different power demands to compare all the results.

Finally, when this is done the controllers should be implemented in the real robot to validate the simulation results and to accomplish the goal of the IUFCV project.

8 REFERENCES

[1] L. Valverde, M. Rosa and C. Bordons, “Gestión de Energía en Sistemas con Fuentes Renovables y almacenamiento de energía basado en Hidrógeno Mediante Control Predictivo,” *Tesis Doctoral,* 2013.

[2] Technopedia, “Technopedia,” [Online]. Available: http://www.technopedia.com/. [Accessed 18 04 2019].

[3] D. Marcos and C. Bordons, “Contributions to Power Management and Dynamics Control in Hybrid Vehicles,” 2014.

[4] D. Feroldi and M. S. Basualdo, “Description of PEM Fuel Cells Systems,” 2012.

[5] California environmental protection agency, “Air Resources Board,” [Online]. Available: https://www.arb.ca.gov/msprog/tech/techreport/fc_tech_report.pdf. [Accessed 15 05 2019].

[6] INTA, US & CSIRO, “IUFCV Porject – Improving efficiency and operational range in unmanned vehicles using fuel cells,” [Online]. Available: http://iufcv.com/. [Accessed 18 01 2019].

[7] Á. Cano and Ridao, “Modelado y control para gestión de potencia en vehículos híbridos,” 2015.

[8] D. Corcoba and C. Bordons, “Control y modelado de vehículo híbrido basado en baterías y supercondensador.,” 2018.

[9] P. Bergman and C. Bordons, “Sistema de supervisión y telemetría para un robot móvil con pila de combustible.,” 2018.

[10] Robotnik, *Summit XL Datasheet.,* vol. a, 2003.

[11] KOKAM, “KOKAM,” [Online]. Available: http://kokam.com/data/Kokam_Cell_Brochure_V.4.pdf. [Accessed 31 05 2019].

[12] BCH, “BCHPOWER,” [Online]. Available: http://www.bchpower.com/en/QingNengDianDui_v109.html. [Accessed 20 05 2019].

[13] V. Cazalla and C. Bordons, “Development of teleoperation and monitoring tools for mobile robots,” 2017.

[14] V. Fernández and F. José, *Sistema de gestión y control,* Huelva, 2018.

[15] “Wikipedia,” [Online]. Available: https://en.wikipedia.org/wiki/Simulink. [Accessed 10 05 2019].

[16] L. Guzella and A. Sciarretta, Vehicle Propulsion Systems. Introduction to modeling and optimization. Third edition., Springer, 2013.

[17] *Control de vehículos eléctricos e híbridos.*

[18] C. Bordons, F. Garcia-Torres and M. A. Ridao, Model Predictive Control of Microgrids, Springer, 2019.

[19] E. F. Camacho and C. Bordons, Model Predictive control, Springer, 2007.

[20] ACEA, “WLTPFACTS,” [Online]. Available: https://wltpfacts.eu/what-is-wltp-how-will-it-work/. [Accessed 17 06 2019].

[21] “Battery University. Basic to Advanced Battery Information.,” 2001. [Online]. Available: http://batteryuniversity.com/. [Accessed 18 01 2019].

[22] C. C. Chan, “The State of the Art of Electric, Hybrid, and Fuel Cell Vehicles,” vol. 95, no. 4, 2007.

[23] E. López, “Almacenamiento y gestión de energía en sistemas híbridos con pila de combustible para vehículos no tripulados.,” 2017.

[24] Robotnik, *Summit XL mobile platform: Hardware Manual.,* vol. b, 2003.

[25] M. Sorensen and S. Kristiansen, “Model Preditive Control for an artificial pancreas,” 2007. [Online]. Available: http://www2.imm.dtu.dk/pubdb/views/edoc_download.php/5313/pdf/imm5313.pdf0. [Accessed 04 06 2019].

[26] Robotnik, *Summit XL mobile platform: Software Manual.,* vol. c, 2003.

[27] Guptagases, “Guptagases,” [Online]. Available: http://www.guptagases.com/guptagases/download/conversion.pdf. [Accessed 12 06 2019].

[28] Energiazero, “Energiazero,” [Online]. Available: http://www.energiazero.org/risparmio_energetico/rolling%20friction%20and%20rolling%20resistance.pdf. [Accessed 22 05 2019].

[29] L. Grüne and J. Pannek, Nonlinear Model Predictive Control, Springer, 2011.

[30] I. Preda, D. Covaciu and C. Gheorghe, “Coast down test - Theoretical and experimental approach,” [Online]. Available: http://aspeckt.unitbv.ro/jspui/bitstream/123456789/7/1/CONAT20104030-Paper.pdf?ncid=txtlnkusaolp00000618. [Accessed 22 05 2019].

[31] L. d. Re, F. Allgöwer, L. Glielmo, C. Guardiola and I. Kolmanovsky, Automotive Model Predictive Control, Springer, 2010.

[32] Ripacciolo, Bernardini, D. Cairano and B. a. Kolmanovsky, “A Stochastic Model Predictive Control Approach for Series Hybrid Electric Vehicle Power Management,” 2010.

[33] L. Wang, Model Predictive Control System Design and Implementation Using MATLAB, Springer, 2009.

52

APPENDIX CODES

init_controllers.m

```
  SOC_INI = 50;
  LOH_INI = 80;
  % Set Real to 1 for WLTP cylce and 0 for real missions
  Real=0;
  %
#############################################################################
########################################

  % Global variables
  % ----------------
      global h            % Stepsize [s] for all other blocks
      global N_sim        % Number of computational steps required to complete
the simulation of the cycle [-] for all other blocks

  % Simulation parameters
  % ---------------------
      h = 1;
      N_sim = 1022;

  %
#############################################################################
########################################


  sample = 1;
  % State Space Model
  microgrid.A = [1 0; 0 1];
  microgrid.B = [1.15e-3 1.15e-3; -16.67 0]*sample;
  microgrid.C = [1 0; 0 1];
  microgrid.D = [0 0; 0 0];
  microgrid.E = [1.15e-3; 0]*sample;


  %% Controller structure
  % Prediciton and control horizons
  controller.Np = 5;
  controller.Nu = 10;

  % Weights in the cost function

  controller.delta = [0.000001 0.000001]; % - Weights for following the
references
  controller.lambda = [.0001 .0001];      % - Weights for the control increment
efforts
  controller.alpha = [.01 0];            % - Weights for the control efforts


  % Definition of the restriction matrices
  % Outputs
  controller.Ymax = [75; 90];%[SOC_max LOH_max][%]
  controller.Ymin = [40; 10];%[SOC_min LOH_min][%]

  % Control signals
```

```
  controller.Umax = [0.2; 0.25];%[Pfc_max-kW Pbat_max-kW]
  controller.Umin = [0; -0.25];%[Pfc_min-kW Pbat_min-kW]

  % Control signal increments
  controller.DeltaUmax = [0.2; 0.25];      %[delta_Pfc_max-kW, delta_Pbat_max-
kW]
  controller.DeltaUmin = [-0.2; -0.25];    %[delta_Pfc_min-kW, delta_Pbat_min-
kW]

  % Initial states
  x0 = [SOC_INI; LOH_INI; 0];           % [SOC-%, LOH-%, Pnet-kW]
  y0 = [SOC_INI; LOH_INI];              % [SOC-%, LOH-%]
  u0 = [0; 0];                          % [Ph2-kW, Pgrid-kW]

  % Reference
  controller.ref = [60; 40];            % [SOc-%, LOH-%]

  % Cost functions and restrictions
  controller.funcCoste = 2;
  controller.restricciones = 1;

  calculo_vector_Pmedia.m
  tiempo=1023;
  Pmedia_ciclo=Pmedia_ciclo_146;
  n=1;
  t=1;
  while t<tiempo
      m=146*n;
      tau=t;
      Pmedia_ciclo(tau)=Pmedia_ciclo_146(m);
      aux=mod(t,146);
      if aux==0
          n=n+1;
      end
      t=t+1;
  end
```

controlHeuristico.m

```
function u=ControlHeristico(a)
% a(1) Power demanded
% a(2) SOC of the battery
% u(1) Power demanded/delivered to the battery
% u(2) Power demanded to the fuel cell
%% Inputs
pdemanded=a(1);
soc=a(2);
%% Variables
SOC_min=40;
SOC_max=75;
SOC_high=60;
SOC_med=50;
P_FC_min=50;
P_FC_med=90;
P_FC_max=180;
P_FC_nom=200;
```

```
%% For positive power demands
if pdemanded >= 0
    % pp=pdemanded;
    if soc > SOC_min && soc <SOC_max
        if pdemanded >= P_FC_min && pdemanded <= P_FC_max
            pp= pdemanded;
            pb=0;
            if soc < SOC_med
                pp=P_FC_max;
                pb=-(P_FC_max-pdemanded);
            end
        else
            if pdemanded > P_FC_max && soc> SOC_high
                pp=P_FC_med;
                pb=pdemanded-pp;
            else
                pp=P_FC_max;
                pb=pdemanded-pp;
            end
            if pdemanded < P_FC_min
                if soc <= SOC_med
                    pp=P_FC_med;
                    pb=-(pp-pdemanded);
                else
                    pp=P_FC_min;
                    pb=pdemanded-pp;
                end
            end
        end
    elseif soc <= SOC_min
        if pdemanded <= P_FC_nom
            pp=P_FC_nom;
            pb=-(pp-pdemanded);
        else
            pp=P_FC_nom;
            pb=0;
        end
    elseif soc >=SOC_max
        pp=0;
        pb=pdemanded;
    end
end
%% For negative power demands (in case of regenerative braking)
% if pdemanded < 0
%     if soc <SOC_med
%         pp=P_FC_max;
%         pb=pdemanded-pp;
%     else
%         pp=P_FC_med;
%         pb=pdemanded-pp;
%     end
% end
%% Outputs
u(1)=pb;
u(2)=pp;
```

controlECMS.m

```
function u=ControlECMS(a)
%init_controllers.m
%a(1) Power demanded
Pdemand=a(1);
%a(2) Schg
%schg=1.2;
schg=a(2);
%a(3) sdis
%sdis=0.69;
sdis=a(3);
%a(4) Actual speed
xdot=a(4);
%a(5) Fuel cell voltage
V_FC=a(5);
%a(6) Previous speed
xdot_prev=a(6);
%a(7) simulation time
simulation_time=a(7);
%a(8) accumulated energy
ac=a(8);
%a(9) State of charge
SOC=a(9);
%u(1) Power demand to the battery
%u(2) Power demand to the fuel cell
%u(3) Actual speed feedback
%u(4) accumulated energy
%u(5) Probability
% F=[];
% T=[];
Jmin=Inf; % Value of the cost function to be updated
%horizon time
th=146;
FC_max_power=200;
m=45;
percentage=0.5;
load potmed
pot_med=[];
if simulation_time>0
    t=round(simulation_time);
    pot_med=Pmedia_ciclo(t);
end
if Pdemand >  0
    if SOC < 40
        pb=0;
        pp=Pdemand;
        xdot_act=xdot;
        Ee_ac=ac;
        Probability=0;
    else
        for i=1:-0.05:0
            pb=i*Pdemand;%power that corresponds to the battery
            pp=(1-i)*Pdemand;%power that corresponds to the fuel cell
```

```
                Ee=ac+(pb+0.5*m*(xdot_prev^2-xdot^2));                    %
Energy

prob=(Ee/(th*FC_max_power))+(pot_med/FC_max_power);%probability of the
battery
                s=sdis*prob+(1-prob)*schg; %s(t) of the battery
                if pp > 1
  %                   intensity=pp/V_FC;
                    x_rend=[0 25 50 75 100 125 150 175 200];
                    y_rend=[0 0.55 0.58 0.66 0.7 0.63 0.55 0.48 0.44];
                    rp=interp1(x_rend,y_rend,pp,'linear');
                else
                    rp= 0.1;
                end
                J=pp/rp+s*pb; % Function cost
                if J < Jmin % if the cost is less than the minimum, updates
the minimum cost
                    Jmin=J; percentage=i;
                end
            end
            pb=Pdemand*percentage;
            pp=Pdemand*(1-percentage);
            xdot_act=xdot;
            Ee_ac=ac+((Pdemand*percentage)+0.5*m*(xdot_prev^2-xdot^2));
            Probability=prob;
        end
  else
        pb= Pdemand;
        pp=0;
        xdot_act=xdot;
        Ee_ac=ac+(Pdemand+0.5*m*(xdot_prev^2-xdot^2));
        Probability=1;
  end
  if pb>250
        pp=pp+(pb-250);
        pb=250;
  elseif pb<-250
        pb=-250;
  end
  if pp>200
        pb=pb+(pp-200);
        pp=200;
  elseif pp<0
        pp=0;
  end
  u(1)=pb;
  u(2)=pp;
  u(3)=xdot_act;
  u(4)=Ee_ac;
  u(5)=percentage;
```

MPC.m

```
  function                 [sys,x0,str,ts,simStateCompliance]                =
MPC(t,x,u,flag,microgrid,controller,SOC_INI,LOH_INI,sample)
  switch flag,
    %% Initialization
    case 0,
      [sys,x0,str,ts,simStateCompliance]                                      =
mdlInitializeSizes(microgrid,SOC_INI,LOH_INI,sample);
    %% Outputs
    case 3,
      sys=mdlOutputs(t,x,u,microgrid,controller);
    %% Non handled cases
      case {1, 2, 4, 9}
      sys = [];
    %% Unexpected flags
    otherwise
      DAStudio.error('Simulink:blocks:unhandledFlag', num2str(flag));
  end

  %%
  % mdlInitializeSizes
  % Return the sizes, initial conditions, and sample times for the S-
function.
  function
[sys,x0,str,ts,simStateCompliance]=mdlInitializeSizes(microgrid,SOC_IN
I,LOH_INI,sample)

      nx = size(microgrid.A,1);                        % Number of states
      ny = size(microgrid.C,1);                        % Number of outputs
      ne  = size(microgrid.B,2);                           % Number of control
signals

      sizes = simsizes;
      sizes.NumContStates   = 0;         % Number of states
      sizes.NumDiscStates   = nx;        % Number of discrete states
      sizes.NumOutputs      = 2;         % Number of outputs
      sizes.NumInputs       = 5;         % Numero de entradas
      sizes.DirFeedthrough  = 1;             % Allow using inputs 'u' to
calculate the outputs in 'mdlOutputs'
      sizes.NumSampleTimes = 1;            % at least one sample time is
needed

      sys = simsizes(sizes);

      x0  = [SOC_INI, LOH_INI];          % States initialization
      str = [];
      ts  = [sample 0];                  % Sample time initialization

      % Specify the block simStateCompliance. The allowed values are:
      %     'UnknownSimState', < The default setting; warn and assume
DefaultSimState
      %    'DefaultSimState', < Same sim state as a built-in block
      %    'HasNoSimState',    < No sim state
```

```
    %     'DisallowSimState' < Error out when saving or restoring the
model sim state
    simStateCompliance = 'UnknownSimState';

% end mdlInitializeSizes

%=======================================================================
=========
% mdlOutputs
% Return the block outputs.
%=======================================================================
=========
%
function sys=mdlOutputs(t,x,u,microgrid,controller)

    %% Read inputs, outputs and disturbances
    % Disturbances
    P_demand = u(1)/1000;    % Power demand [kW]
    % Outputs
    SOC_act = u(2);          % State of Charge [%]
    LOH_act = u(3);          % Level of hydrogen [%]
    % Control inputs
    Pfc = u(4)/1000;              % Fuel cell power [kW]
    Pbat = u(5)/1000;             % Battery power [kW]


    % Definition of aumented matrices
    A = [microgrid.A microgrid.E; zeros(1,2) eye(1)];
    B = [microgrid.B; zeros(1,2)];
    C = [microgrid.C zeros(2,1)];




    % Size of the process
    nx = size(A,1);          % Number of states
    ny = size(C,1);          % Number of outputs
    ne = size(B,2);          % Number of control signals

    % Definition of the incremental state space model
    % Aumented state space: xam(k) = [x(k); u(k-1)]
    % - Definition of matrices
    M = [A B; zeros(ne,nx) eye(ne)];
    N = [B; eye(ne)];
    Q = [C zeros(ny,ne)];

    %% COST FUNCTION WEIGHTS MATRICES
    % SIZE deltaM = zeros(Np*ny, Np*ny);
    delta1 = [];
    for i = 1:1:controller.Np
        delta1 = [delta1 controller.delta];
    end
    deltaM = diag(delta1);

    lambda1 = [];
    % SIZE lambdaM = zeros(Nu*ne, Nu*ne);
    for i = 1:1:controller.Nu
```

```
    lambda1 = [lambda1 controller.lambda];
end
lambdaM = diag(lambda1);

alpha1 = [];
% SIZE alphaM = zeros(Nu*ne, Nu*ne);
for i = 1:1:controller.Nu
    alpha1 = [alpha1 controller.alpha];
end
alphaM = diag(alpha1);

%% CALC OF THE PREDICTION
% - Matrix F
F = [];
for i = 1:1:controller.Np
    F = [F; Q*M^(i)];
end

H = [];
zerosH = zeros(size(Q*N));
for i=1:1:controller.Np
    hLine = [];
    for j=1:1:controller.Nu
        if j<=i
            eme = M^(i-j);
            hLine = [hLine Q*eme*N];
        else
            hLine = [hLine zerosH];
        end
    end
    H = [H;hLine];
end

%% RESTRICTIONS
B1=[];
B2=[];
for i=1:1:controller.Nu
    B1 = [B1; controller.DeltaUmax];
    B2 = [B2; -controller.DeltaUmin];
end

% Id = eye(Nu*ne,Nu*ne);
YMaxArray = [];
YMinArray = [];
for i = 1:1:controller.Np
   YMaxArray = [YMaxArray;controller.Ymax];
   YMinArray = [YMinArray;controller.Ymin];
end

UMaxArray = [];
UMinArray = [];
for i = 1:1:controller.Nu
   UMaxArray = [UMaxArray;controller.Umax];
   UMinArray = [UMinArray;controller.Umin];
end
```

```
    %T = tril(ones(Nu*ne,Nu*ne));% - diag([1 1 1 1]);
    T=[];
    miniT=eye(ne);
    miniZeros=zeros(ne);
    for i=1:1:controller.Nu
        tLine = [];
        for j=1:1:controller.Nu
            if j<=i
                tLine = [tLine, miniT];
            else
                tLine = [tLine, miniZeros];
            end
        end
        T=[T;tLine];
    end

%**********************************************************************
**
    %% INITIAL CONDITIONS

    x = [SOC_act; LOH_act; P_demand];

    y = [SOC_act; LOH_act];

    ut = [Pfc; Pbat];

    %% MATRIX OF REFERENCES
    w = zeros(ny*controller.Np,1);

    for i = 1:ny:controller.Np*ny
        w(i:i+ny-1,1) = controller.ref;
    end
    %% MATRIX CONTROL SIGNALS T-1
    utM = ones(ne*controller.Nu,1);
    for j = 1:ne:ne*controller.Nu
        utM(j:j+ne-1,1) = ut;
    end

  %  % Sampling
  %  xt = controller.A*x(1:2) + controller.B*ut + controller.E*x(3);
  %  y = controller.C*x(1:2);

    xam = [x; ut];

%**********************************************************************
**

    %% Matrices for QP
    if controller.funcCoste == 1                                      % J
= (y-w)'*deltaM*(y-m) + DeltaU'*lambdaM*DeltaU
        Hqp = 2*(H'*deltaM*H + lambdaM);
        Bqp = 2*(F*xam-w)'*deltaM*H;
```

```
    elseif controller.funcCoste == 2                                 % J
= (y-w)'*deltaM*(y-m) + DeltaU'*lambdaM*DeltaU + U'*AlphaM*U
        Hqp = 2*(H'*deltaM*H + lambdaM + T'*alphaM*T);
        Bqp = 2*(F*xam-w)'*deltaM*H + 2*utM'*alphaM*T;
    else                                                             % J
= (y-w)'*deltaM*(y-m) + U'*AlphaM*U
        Hqp = 2*(H'*deltaM*H + T'*alphaM*T);
        Bqp = 2*(F*xam-w)'*deltaM*H + 2*utM'*alphaM*T;
    end

    % Matrix for QP: Aqp
    A1 = eye(controller.Nu*ne,controller.Nu*ne);
    A2 = -A1;
    A3 = H;
    A4 = -A3;
    A5 = T;
    A6 = -A5;

    Aqp =[A1; A2; A3; A4; A5; A6];

    % Restriction matrices QP: Bqp
    B3 = YMaxArray - F*xam;
    B4 = F*xam - YMinArray;

    utArray = [];
    for j = 1:1:controller.Nu
       utArray = [utArray;ut];
    end

    B5 = UMaxArray - utArray;
    B6 = utArray - UMinArray;

    Brqp = [B1;B2;B3;B4;B5;B6];

    %options   =   optimoptions('quadprog','Algorithm','interior-point-
convex','Display','off','MaxIterations',15000);
    if controller.restricciones==1
        %DeltaU = quadprog(Hqp,Bqp,Aqp,Brqp,[],[],[],[],[],options);
        DeltaU = quadprog(Hqp,Bqp,Aqp,Brqp);
    else
        %DeltaU = quadprog(Hqp,Bqp,[],[],[],[],[],[],[],options);
        DeltaU = quadprog(Hqp,Bqp);
    end

    % Update the value of the control signal
    u = ut + DeltaU(1:ne,1);

    Pfc = u(1);
    Pbat = P_demand-Pfc;

    sys = [];
    sys = [Pbat*1000 Pfc*1000];

  % end mdlOutputs
```

PLOT.m

```
  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 %% HEURISTIC

 % Power
 figure(1);
 plot(P_heuristic);
 title('Power demand vs Battery and Fuel cell supply in heuristic
control strategy');
 grid ON;
 xlim([0 1022]);
 ylabel('Power(V)');
 legend({'Power demand','Battery Power','Fuel cell Power'});

 %Level of charge
 figure(2);
 plot(Level_of_charge_heuristic);
 title('Levels of charge heristic');
 grid ON;
 xlim([0 1022]);
 ylabel('Level of charge (%)');
 ylim([0 100]);
 legend({'SOC','LOH'});
 inc_SOC_heuristic=Level_of_charge_heuristic.data(1023)-
Level_of_charge_heuristic.data(1);
 inc_LOH_heuristic=Level_of_charge_heuristic.data(2046)-
Level_of_charge_heuristic.data(1024);


 %% ECMS

 % Power
 figure(3);
 plot(P_ECMS);
 title('Power demand vs Battery and Fuel cell supply in ECMS control
strategy');
 grid ON;
 xlim([0 1022]);
 ylabel('Power(V)');
 legend({'Power demand','Battery Power','Fuel cell Power'});

 %Level of charge
 figure(4);
 plot(Level_of_charge_ECMS);
 title('Levels of charge ECMS');
 grid ON;
 xlim([0 1022]);
 ylabel('Level of charge (%)');
 ylim([0 100]);
 legend({'SOC','LOH'});
 inc_SOC_ECMS=Level_of_charge_ECMS.data(1023)-
Level_of_charge_ECMS.data(1);
 inc_LOH_ECMS=Level_of_charge_ECMS.data(2046)-
Level_of_charge_ECMS.data(1024);
```

```
%% MPC

% Power
figure(5);
plot(P_MPC);
title('Power demand vs Battery and Fuel cell supply in MPC control strategy');
grid ON;
xlim([0 1022]);
ylim([-250 400]);
ylabel('Power(V)');
legend({'Power demand','Battery Power','Fuel cell Power'});

%Level of charge
figure(6);
plot(Level_of_charge_MPC);
title('Levels of charge MPC');
grid ON;
xlim([0 1022]);
ylabel('Level of charge (%)');
ylim([0 100]);
legend({'SOC','LOH'});
inc_SOC_MPC=Level_of_charge_MPC.data(1023)-
Level_of_charge_MPC.data(1);
inc_LOH_MPC=Level_of_charge_MPC.data(2046)-
Level_of_charge_MPC.data(1024);
```